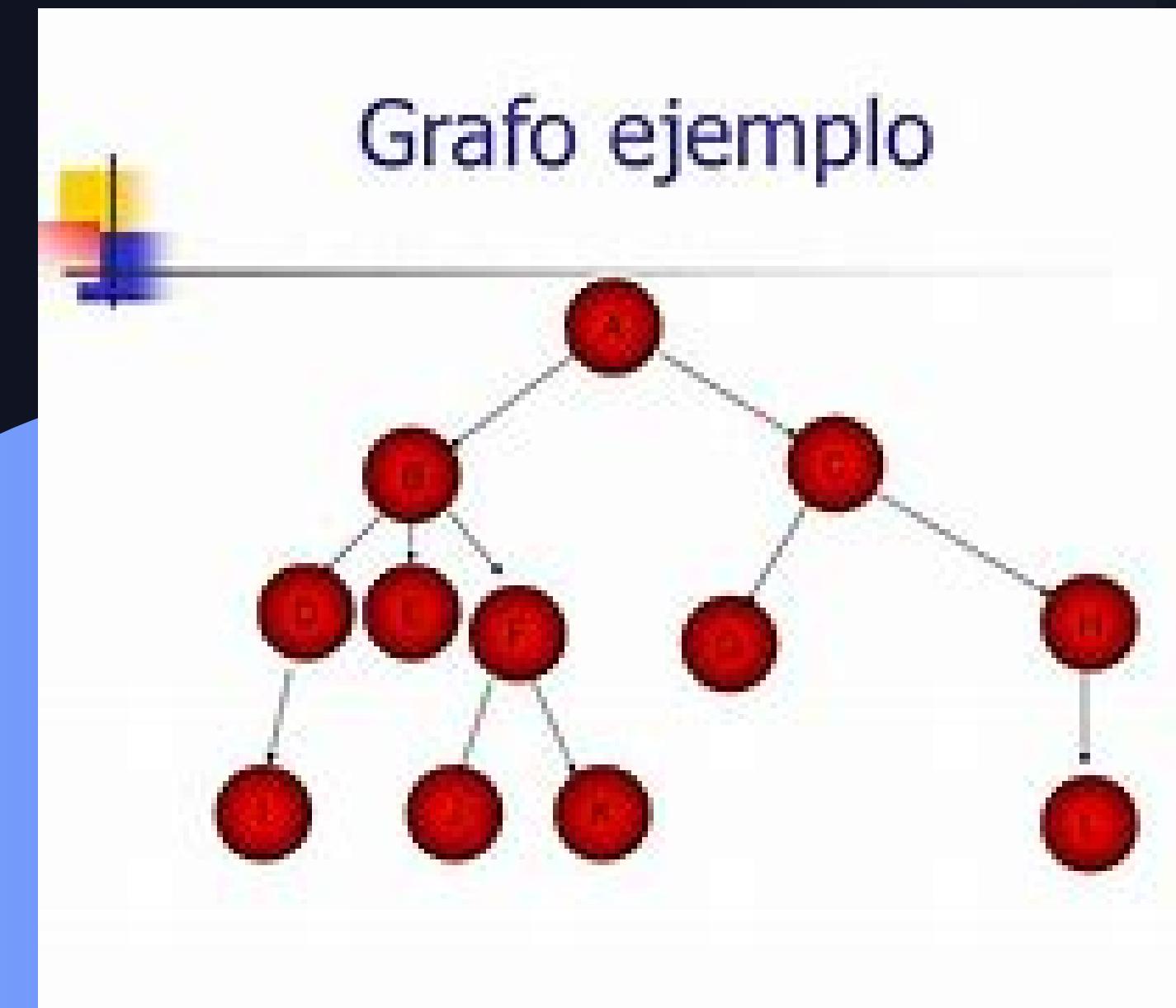
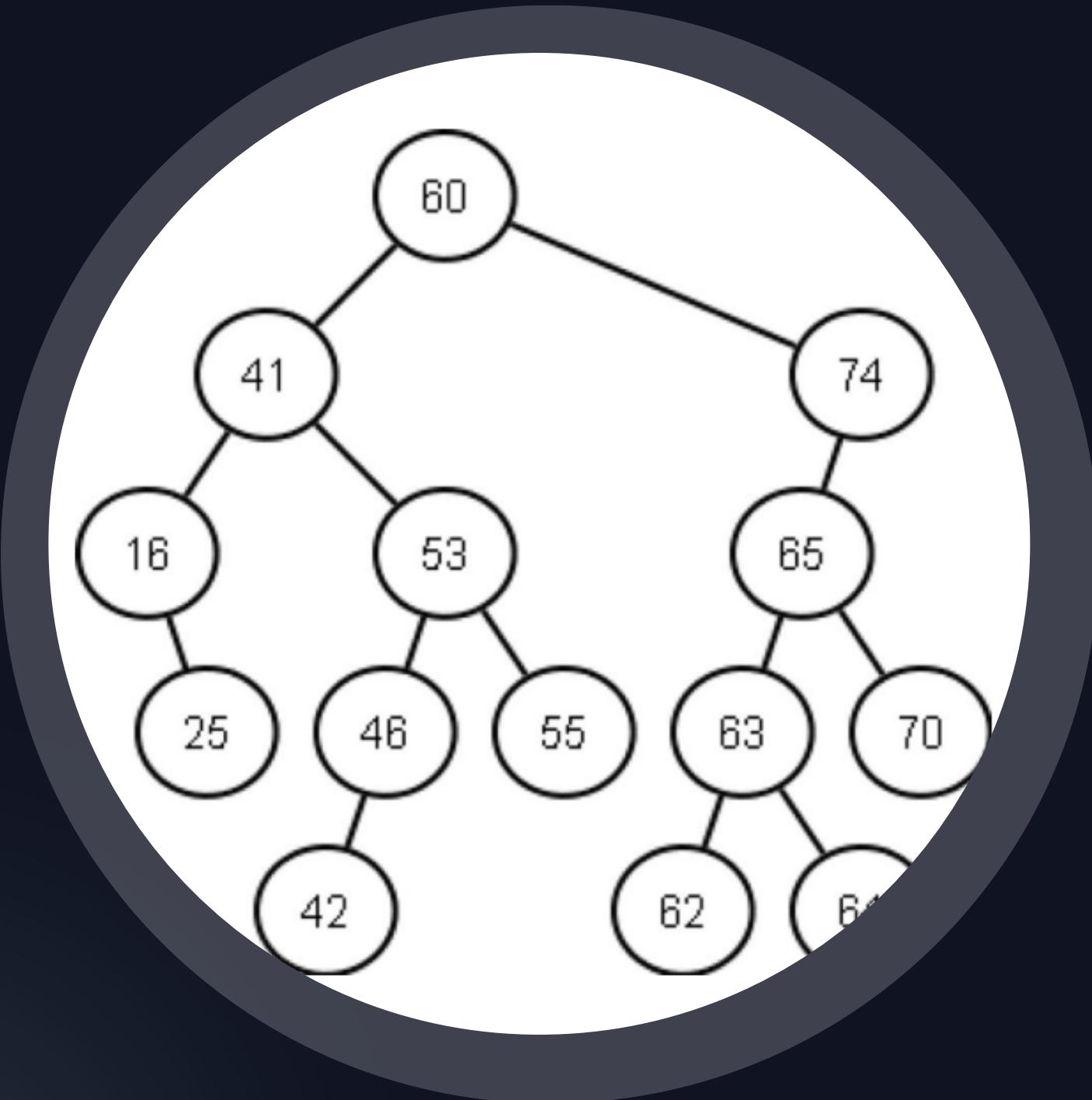


ARBOLES

- INTEGRANTES:
PEDRO SENTÍES ROBLES
CESAR RIOS MENDEZ
BRAYAN HERNANDEZ ACEVEDO
PEDRO TEZOCHO CRUZ





ARBOLES

Un árbol se puede definir como una estructura jerárquica y en forma no lineal, aplicada sobre una colección de elementos u objetos llamados nodos.

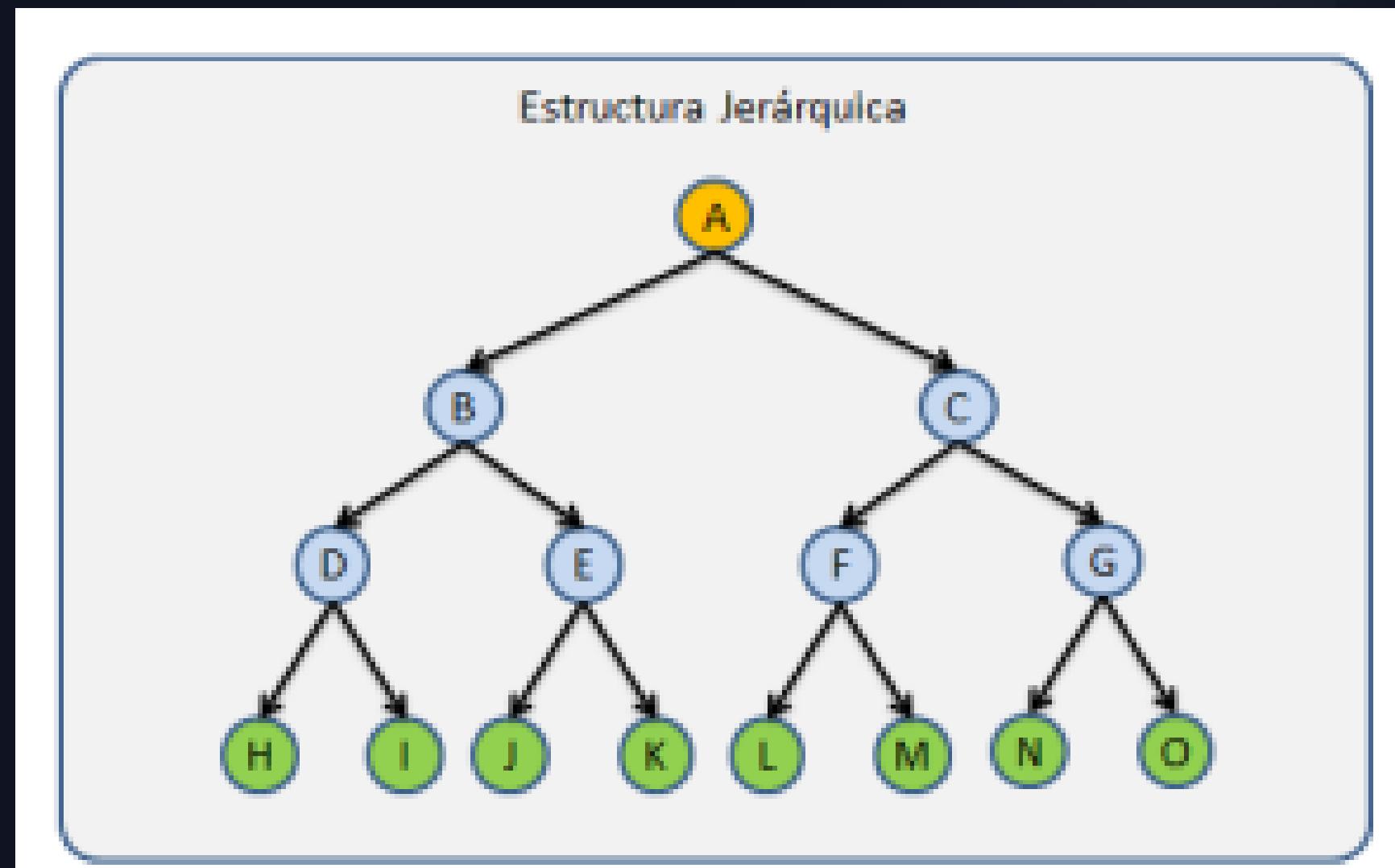


ÁRBOLES

Los árboles son considerados las estructuras de datos no lineales y dinámicas de datos muy importantes del área de computación.

Los árboles son muy utilizados en informática como un método eficiente para búsquedas grandes y complejas.

Casi todos los sistemas operativos almacenan sus archivos en árboles o estructuras similares a árboles.



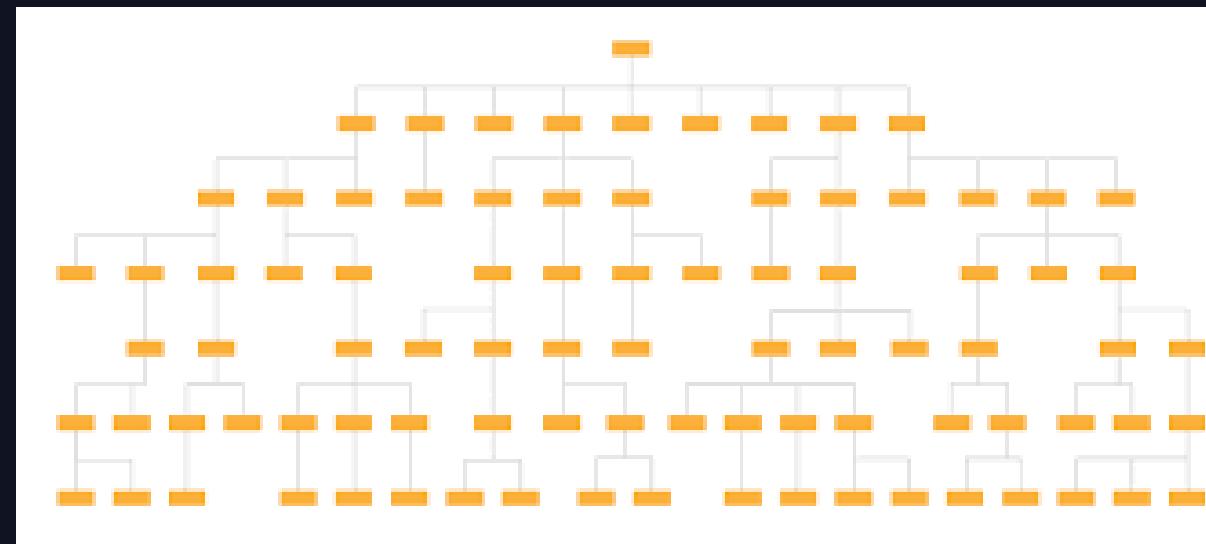


FORMAS DE REPRESENTAR UN ARBOL

1. Representación textual: Esta forma utiliza una notación textual para representar la estructura del árbol. Por ejemplo, se puede utilizar paréntesis y comas para indicar las ramas y subramas. Por ejemplo, un árbol binario con raíz "A", hijo izquierdo "B" y hijo derecho "C" se puede representar como "(A, (B, ,), (C, ,))".



2. Diagrama: Es una representación gráfica del árbol utilizando nodos y enlaces. Cada nodo representa un elemento del árbol, y los enlaces representan las relaciones jerárquicas entre los nodos. Los nodos pueden estar etiquetados con los valores correspondientes. Por ejemplo:

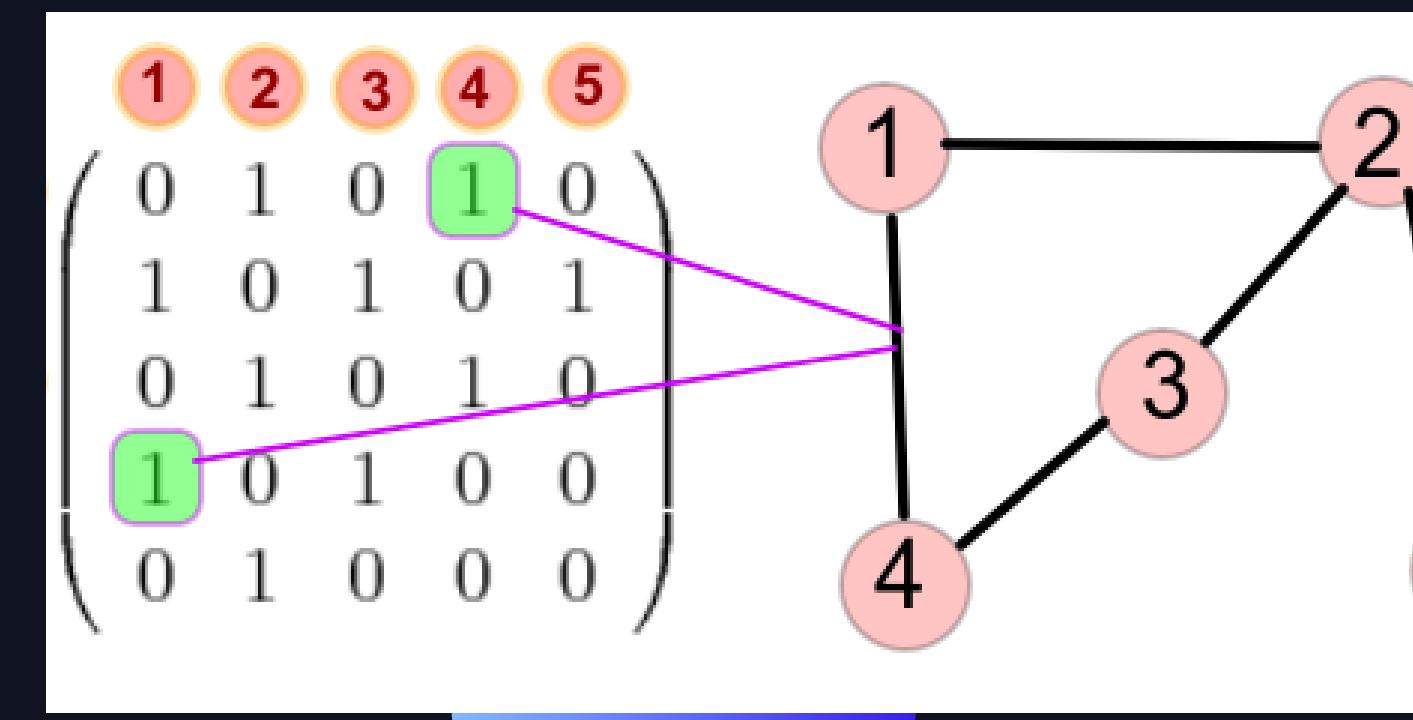




3. Lista de listas: En esta representación, se utiliza una lista de listas para representar la estructura del árbol. Cada lista interior representa un nodo, y los elementos de la lista interior representan los hijos del nodo. Por ejemplo, el árbol binario "A" -> "B", "C" se puede representar como `["A", ["B", []], ["C", []]]`.



4. Matriz de adyacencia: Esta forma de representación utiliza una matriz booleana para indicar las conexiones entre los nodos del árbol. Si hay una conexión entre dos nodos, el valor correspondiente en la matriz será verdadero (o 1); de lo contrario, será falso (o 0). Esta representación es útil para árboles en los que la posición de los nodos es importante. Por ejemplo, para el árbol binario "A" -> "B", "C", la matriz de adyacencia sería:





5. Lista de enlaces: En esta representación, se utiliza una lista de enlaces para representar las conexiones entre los nodos del árbol. Cada enlace tiene dos elementos: el nodo origen y el nodo destino. Puede ser una representación eficiente en términos de almacenamiento si el árbol no está completamente balanceado. Por ejemplo, el árbol binario "A" -> "B", "C" se puede representar como [("A", "B"), ("A", "C")].

CARACTERISTICAS DE UN ARBOL

Grafos, anidación de paréntesis y diagramas de venn.

Nodo Hijo: Es aquel nodo que siempre va a tener un nodo antecesor o padre, son aquellos que se encuentran en el mismo nivel

Padre: Es aquel que tiene hijos y también puede tener o no antecesores.

Hermano: Dos nodos son hermanos si son apuntados por el mismo nodo, es decir si tienen el mismo padre.

Raíz: Es el nodo principal de un árbol y no tiene antecesores.

Hoja o terminal: Son aquellos nodos que no tienen hijos o también los nodos finales de un árbol.

Interior: Se dice que un nodo es interior si no es raíz ni hoja.

Nivel de un nodo: Se dice que el nivel de un nodo es el numero de arcos que deben ser recorridos, partiendo de la raíz para llegar hasta el.

Altura del árbol: Se dice que la altura de un árbol es el máximo de los niveles considerando todos sus nodos.

Grado de un nodo: se dice que el grado de un nodo es el número de hijos que tiene dicho nodo.



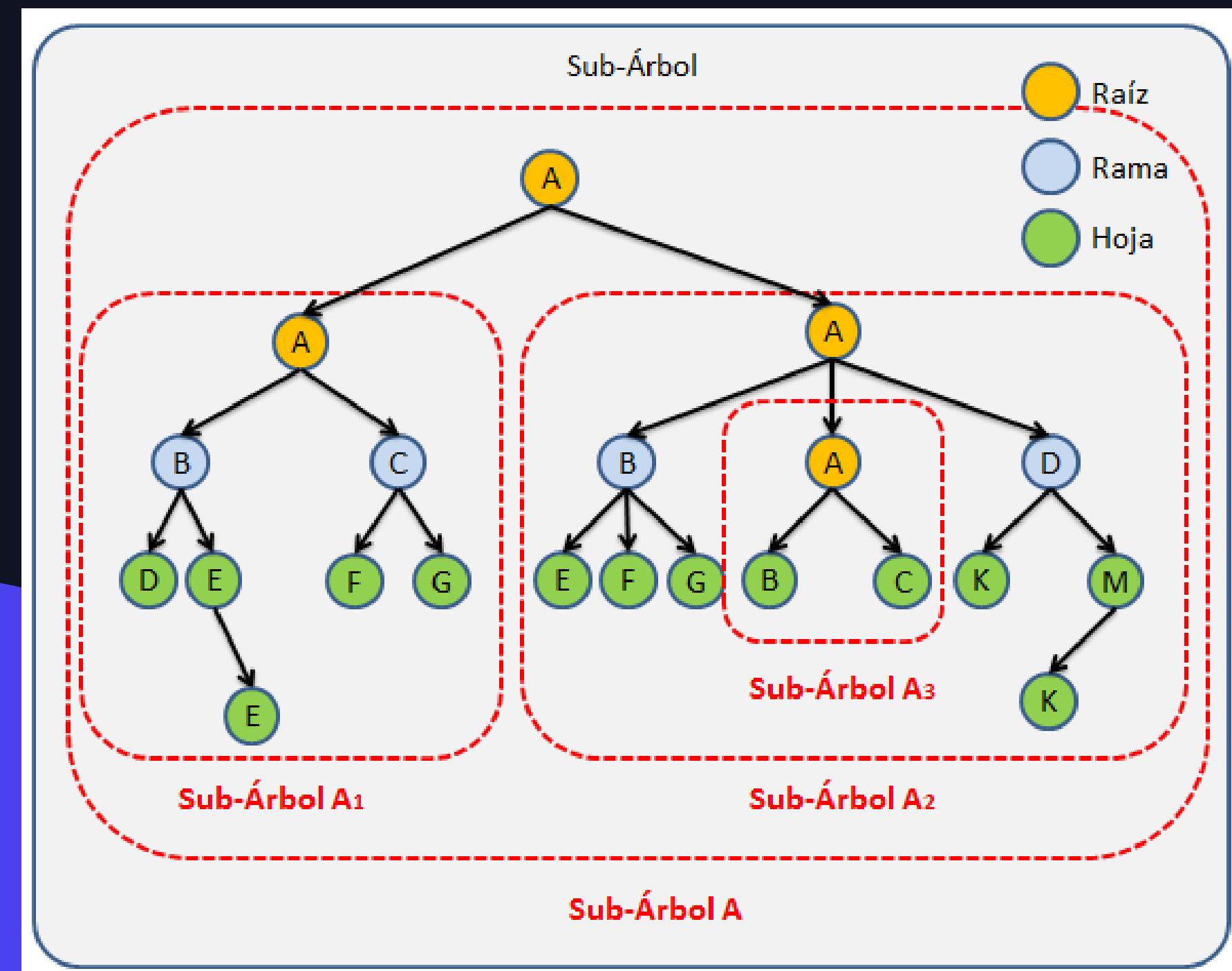
4) QUE ES UN BOSQUE

UN BOSQUE

En estructura de datos, un bosque se refiere a una colección de árboles. Un árbol en estructura de datos es una estructura no lineal compuesta por nodos interconectados. Cada árbol en un bosque es independiente y tiene su propia raíz, ramas y hojas.

En términos más técnicos, un bosque en estructura de datos se basa en la estructura de árbol y se utiliza comúnmente en algoritmos de aprendizaje automático, como el algoritmo de bosques aleatorios (random forests). En el contexto del aprendizaje automático, un bosque se crea al combinar múltiples árboles de decisión en un solo modelo.

En un bosque aleatorio, cada árbol se entrena de forma independiente utilizando diferentes subconjuntos de datos y características seleccionadas al azar. Luego, los resultados de los árboles individuales se combinan mediante votación o promediado para producir una predicción final



El uso de un bosque en estructura de datos, como los bosques aleatorios, puede proporcionar mejoras en la precisión y la estabilidad del modelo, al reducir el sobreajuste y aprovechar la diversidad de los diferentes árboles que lo componen.

En resumen, un bosque en estructura de datos es una colección de árboles interconectados que se utiliza para realizar operaciones y cálculos eficientes, especialmente en el contexto del aprendizaje automático.

5) REGLAS PARA BALANCAR UN ARBOL GENERAL A UNO BINARIO

Al convertir un árbol general en un árbol binario, se deben seguir ciertas reglas para mantener la estructura y el equilibrio del árbol. las reglas principales para balancear un árbol general a uno binario:

1. Selección del nodo raíz: Elija un nodo del árbol general para que sea la raíz del árbol binario resultante. Puede elegir cualquier nodo del árbol general como raíz, pero es recomendable seleccionar un nodo que tenga una estructura y contenido significativos para mantener la semántica del árbol original.

2^o. Asignación de hijos: Asigne los hijos directos del nodo raíz del árbol general como hijos izquierdo y derecho del nodo raíz en el árbol binario.

3. Conversión de nodos internos: Para cada nodo interno restante en el árbol general (es decir, aquellos que no son el nodo raíz ni los hijos directos del nodo raíz), cree un nuevo nodo en el árbol binario y asigne el nodo original como hijo izquierdo del nuevo nodo. Si el nodo tiene más de un hijo, elija uno de los hijos como hijo derecho y convierta los hijos restantes en subárboles anidados dentro del hijo derecho.

4.⁺ Eliminación de nodos innecesarios: Si hay nodos en el árbol binario que solo tienen un hijo (ya sea izquierdo o derecho) y no tienen hijos adicionales, puede eliminar esos nodos y enlazar directamente su único hijo con el padre correspondiente. Esto ayuda a simplificar la estructura del árbol binario.

Es importante tener en cuenta que el equilibrio de un árbol binario puede depender de las reglas específicas del problema y del contexto en el que se esté trabajando. Por lo tanto, las reglas mencionadas anteriormente son generales y pueden requerir ajustes dependiendo de los requisitos y restricciones específicos de la aplicación.

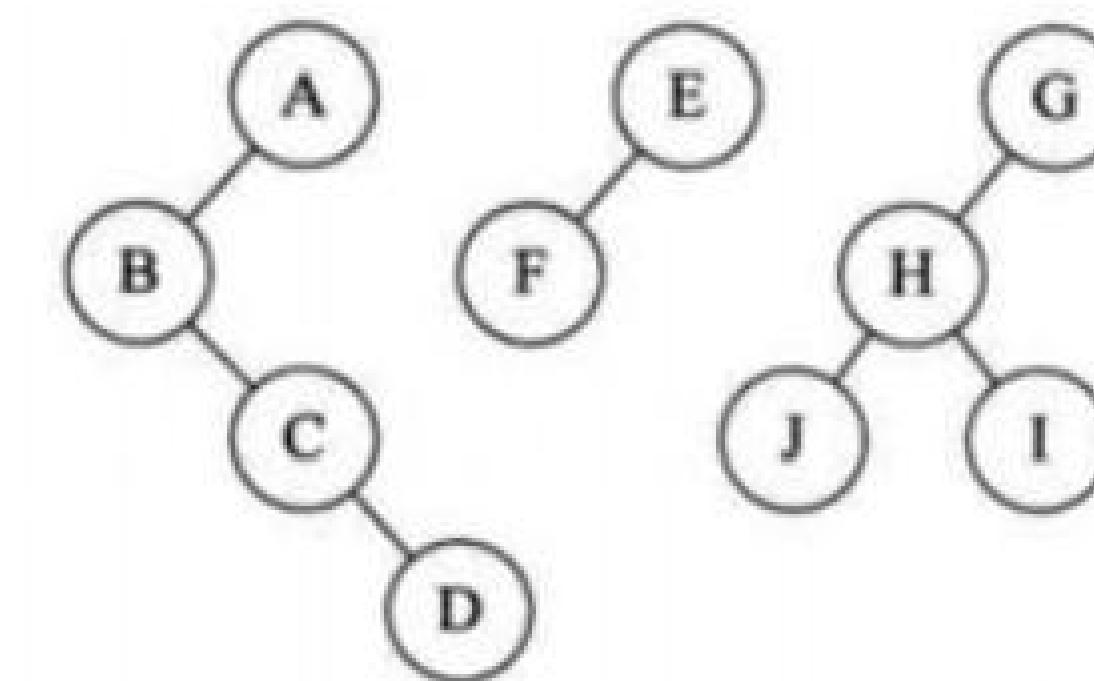
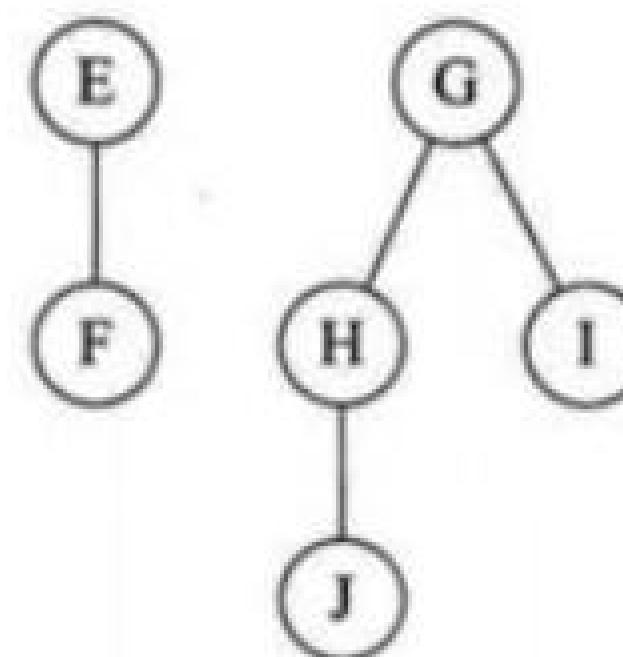
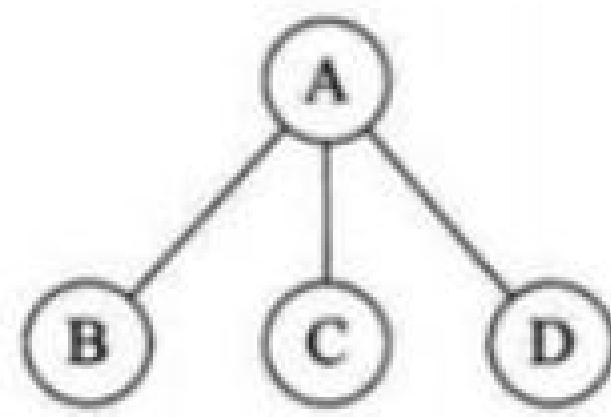
6) REGLAS PARA BALANCEAR UN BOSQUE A ARBOL BINARIO

El bosque está compuesto por varios árboles, se puede entender que cada árbol en el bosque es un hermano y se puede operar de acuerdo con el tratamiento del hermano.

Los pasos son los siguientes:

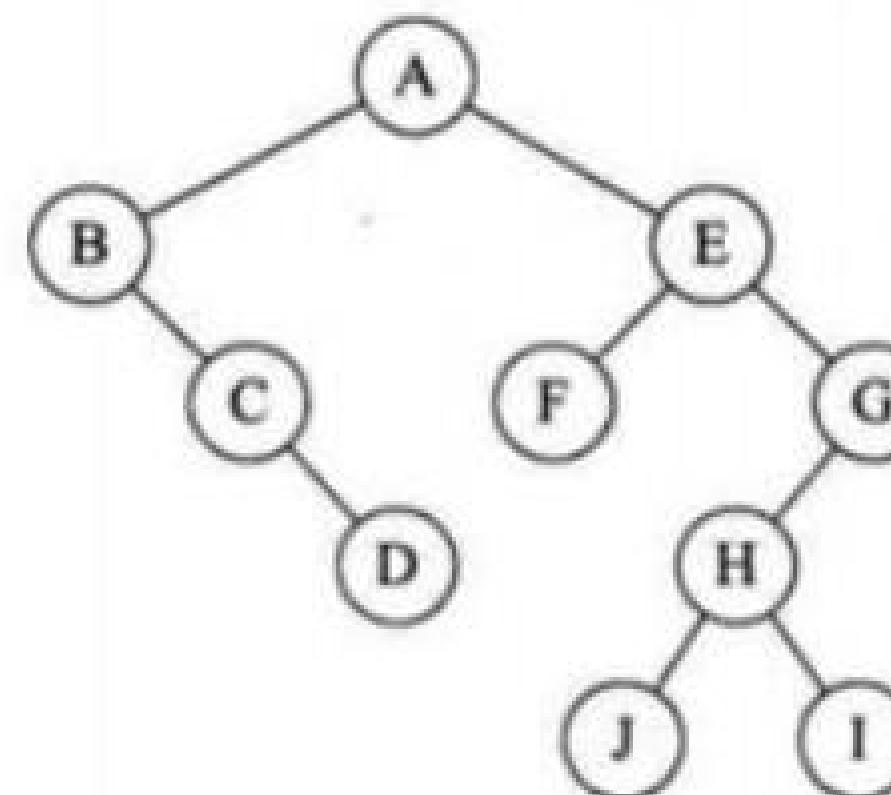
- 1. Convierte cada árbol en un árbol binario.**
- 2. El primer árbol binario no se mueve. A partir del segundo árbol binario, ponga.**

El hijo derecho del nodo raíz del último árbol binario es el nodo raíz del árbol binario anterior., Conectado con cables. Cuando todos los árboles binarios están conectados, se obtiene el árbol binario convertido del bosque.



拥有三棵树的森林

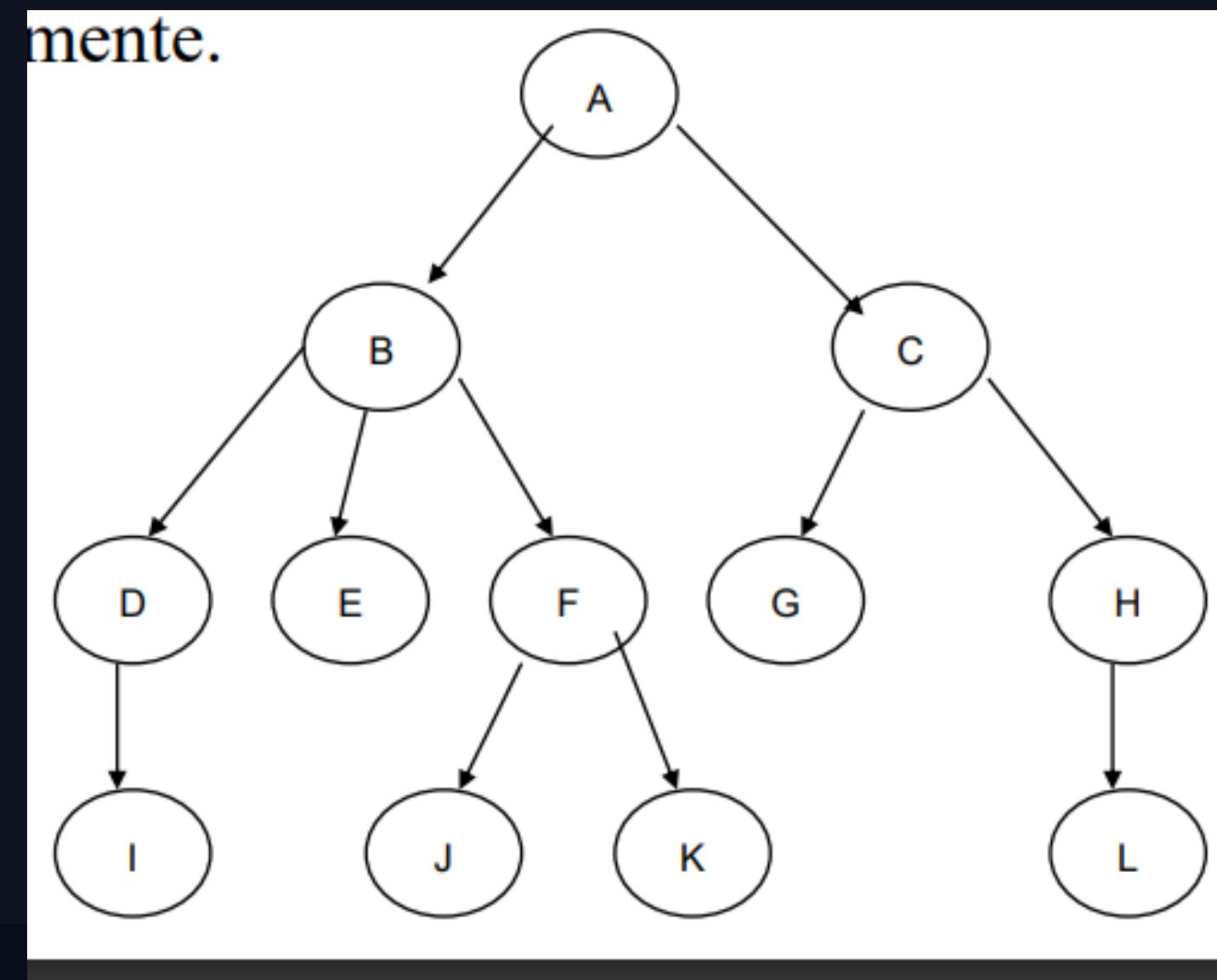
步骤1: 森林中每棵树转换为二叉树



步骤2: 将所有二叉树转换为一棵二叉树

7) LONGITUD DE CAMINO INTERNO Y EXTERNO.

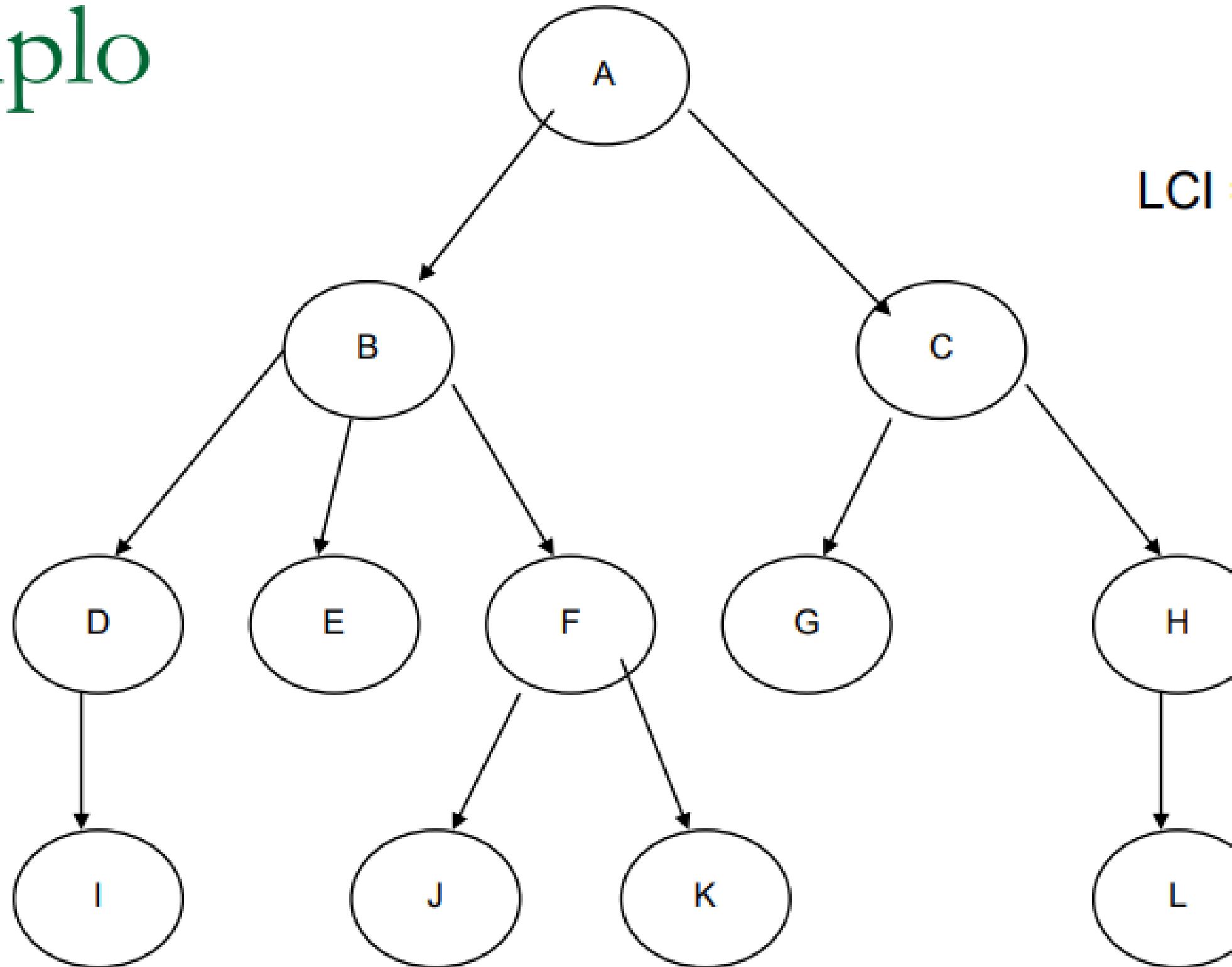
Se define la longitud de camino X como el número de arcos que deben ser recorridos para llegar desde la raíz al nodo X. Por definición la raíz tiene longitud de camino 1, sus descendientes directos longitud de camino 2 y así sucesivamente.



La longitud de camino interno es la suma de las longitudes de camino de todos los nodos del árbol. Es importante porque permite conocer los caminos que tiene el árbol. Puede calcularse por medio de la siguiente fórmula: donde „i“ representa el nivel del árbol, „h“ su altura y „ni“ el número de nodos en el nivel „i“.

$$LCI = \sum_{i=1}^h ni * i$$

Ejemplo



$$LCI = \sum_{i=1}^h n_i * i$$

La LCI del árbol anterior es: $LCI = 1*1 + 2*2 + 5*3 + 4*4 = 36$
 $h=4$

Longitud de camino externo:

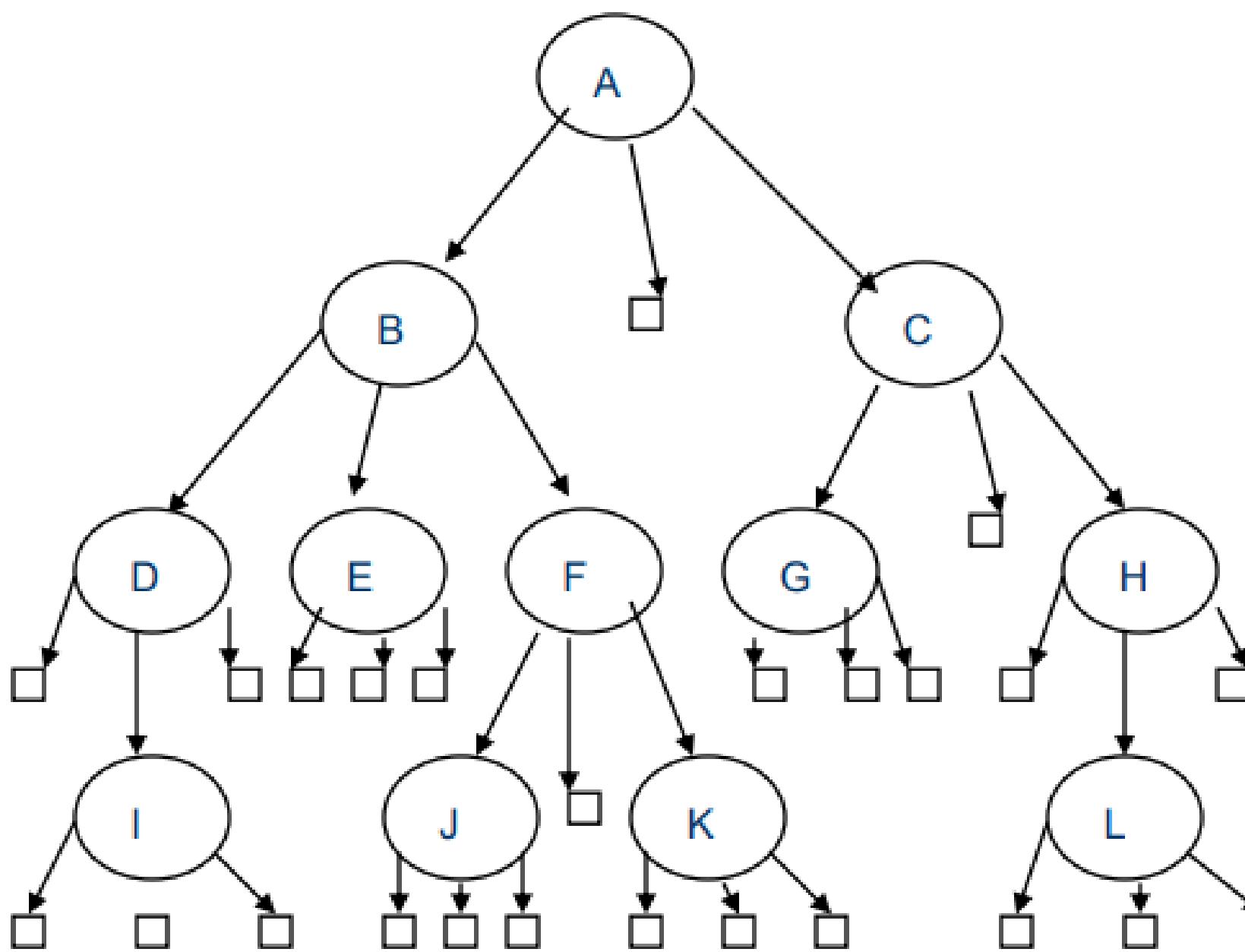
Primero definiremos los conceptos de:

Árbol extendido es aquel en el que el número de hijos de cada nodo es igual al grado del árbol. Si alguno de los nodos del árbol no cumple con esta condición entonces debe incorporársele al mismo nodos especiales; tantos como sea necesario para satisfacer la condición.

Los nodos especiales tienen como objetivo reemplazar las ramas vacías o nulas, no pueden tener descendientes y normalmente se representan con la forma de un cuadrado.

Ejemplo

El número de nodos especiales de este árbol es 25, del árbol de grado 3.

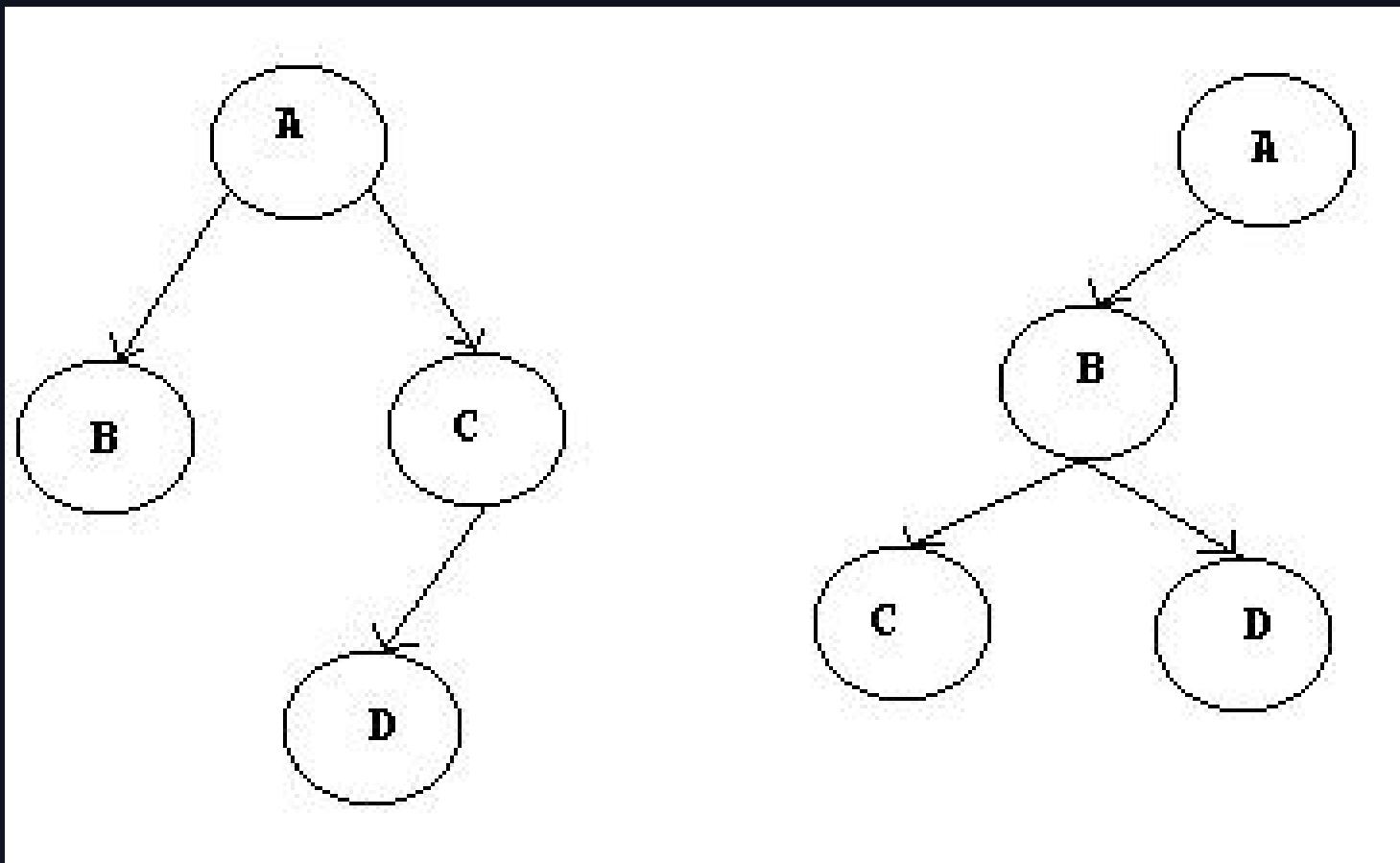


TIPOS DE ARBOLES: ARBOLES BINARIOS DISTINTOS, SIMILARES Y EQUIVALENTES , ETC

Arboles binarios distintos

Un árbol binario es un nodo externo, o uno interno conectado a un par de árboles binarios, llamados el sub-árbol derecho y el sub-árbol izquierdo.

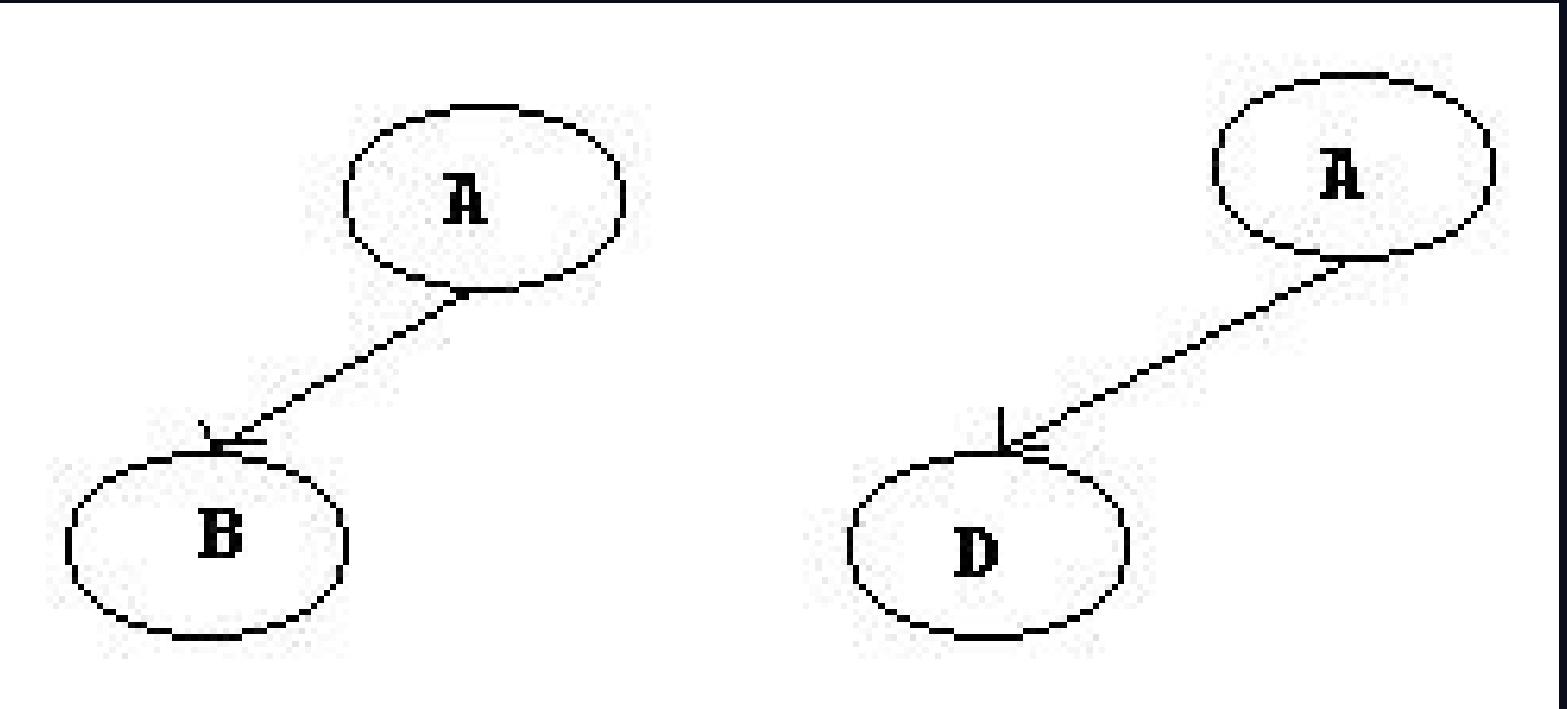
Dos árboles binarios se consideran distintos si su estructura difiere, es decir, si tienen diferentes disposiciones de nodos y conexiones, incluso si sus contenidos son idénticos. Por ejemplo, si un árbol tiene un nodo raíz con un hijo izquierdo y otro árbol tiene un nodo raíz con un hijo derecho, se considerarían árboles binarios distintos.



Arboles binarios similares

Dos árboles binarios son distintos cuando sus estructuras son idénticas, pero la información que contienen sus nodos, difiere entre sí.

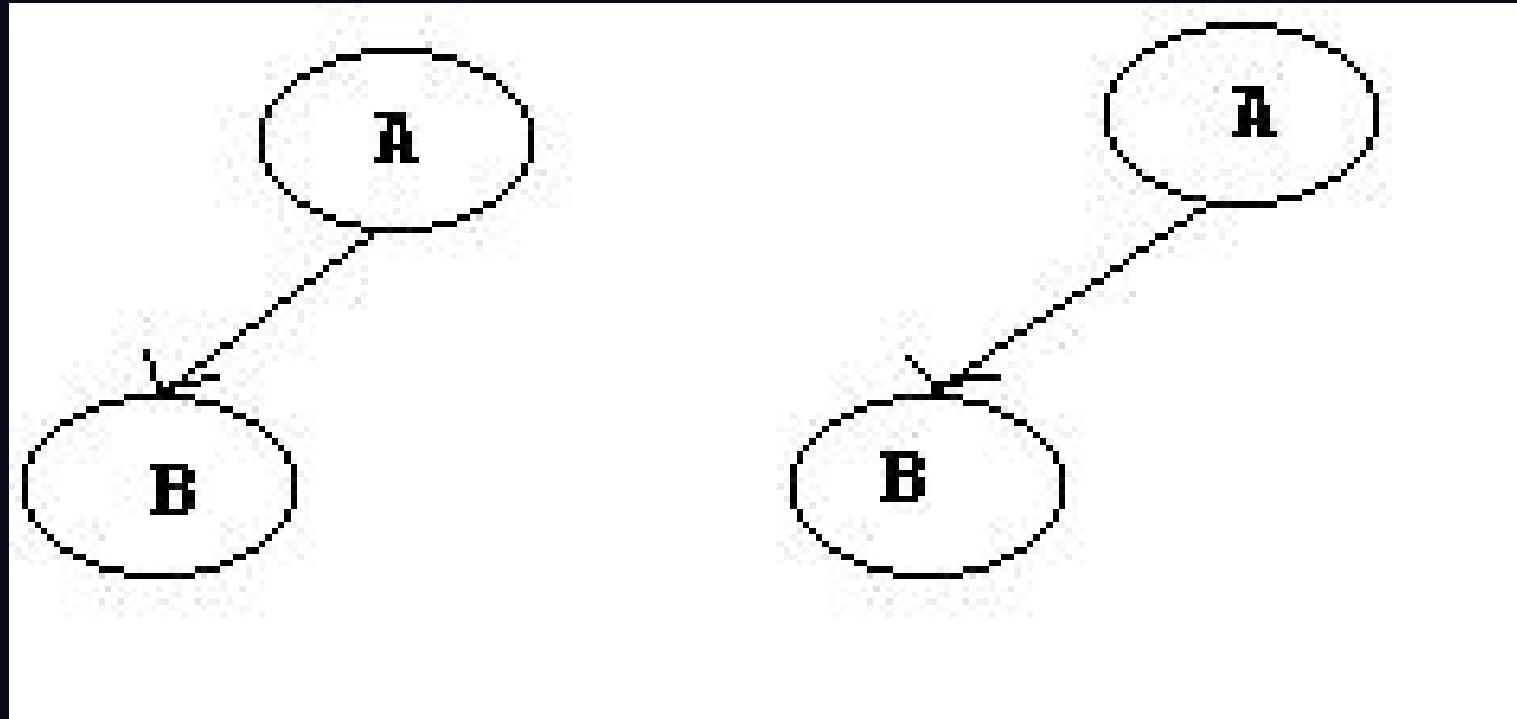
Dos árboles binarios se consideran similares si tienen la misma estructura, aunque sus contenidos pueden ser diferentes. Esto significa que los nodos y las conexiones son idénticos, pero los valores almacenados en los nodos pueden variar. Por ejemplo, si dos árboles tienen la misma disposición de nodos y conexiones, pero los valores almacenados en los nodos son diferentes, se considerarían árboles binarios similares.



Arboles binarios equivalentes

Los árboles binarios equivalentes se definen como aquellos que son similares y además los nodos contienen la misma información.

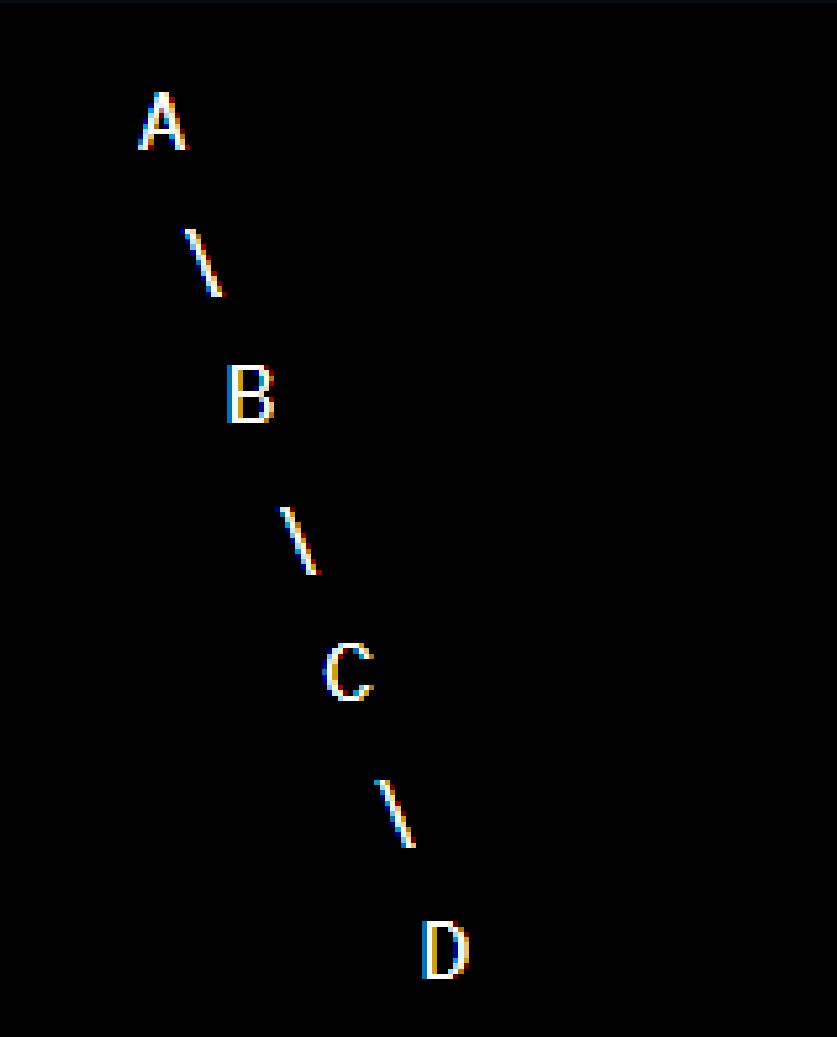
Dos árboles binarios se consideran equivalentes si tienen la misma estructura y los mismos contenidos en cada uno de sus nodos. Esto significa que los nodos, las conexiones y los valores almacenados en los nodos son idénticos en ambos árboles. Por ejemplo, si dos árboles tienen la misma disposición de nodos y conexiones, y los valores almacenados en los nodos son iguales, se considerarían árboles binarios equivalentes.



Árbol binario degenerado

Es un árbol binario en el que cada nodo tiene como máximo un hijo. Es esencialmente una lista enlazada.

Un árbol binario degenerado es un tipo especial de árbol binario en el que cada nodo tiene como máximo un hijo. Es decir, es esencialmente una lista enlazada. En un árbol binario degenerado, todos los nodos tienen solo un hijo, ya sea el hijo izquierdo o el hijo derecho, o ninguno si es un nodo hoja.

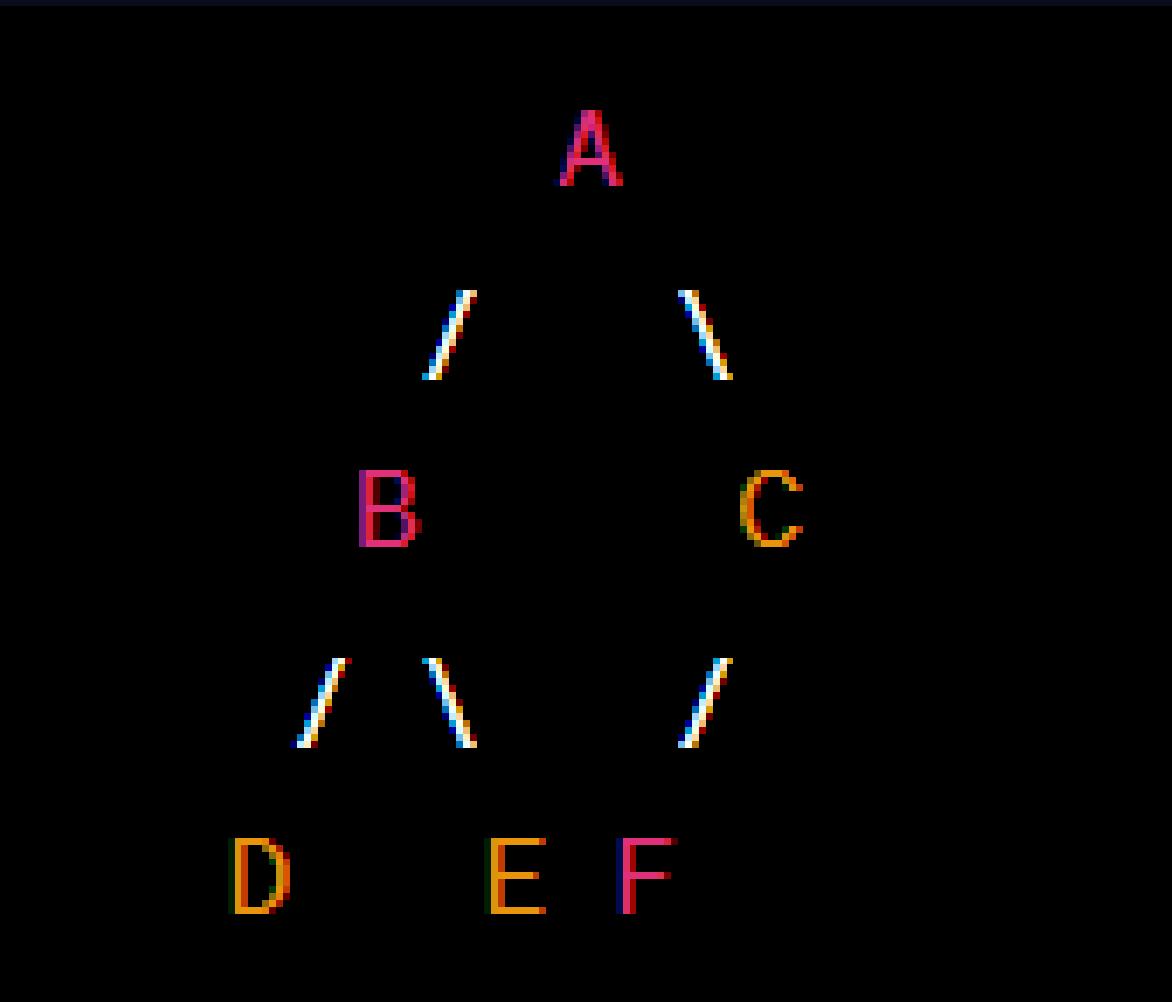


• ARBOL BINARIO COMPLETO.

Se define un árbol binario completo como un árbol en el que todos sus nodos, excepto los del último nivel, tienen dos hijos; el subárbol izquierdo y el subárbol derecho.

Un árbol binario completo es un tipo de árbol binario en el que todos los niveles, excepto posiblemente el último, están completamente llenos, y los nodos en el último nivel están ubicados lo más a la izquierda posible. Esto significa que todos los nodos tienen exactamente cero o dos hijos.

En este ejemplo, todos los niveles están completamente llenos, excepto el último nivel. Los nodos D y E están ubicados lo más a la izquierda posible en el último nivel. Cada nodo tiene cero o dos hijos, cumpliendo así la propiedad de un árbol binario completo.



EJEMPLOS DE APLICACIÓN DE ARBOLES.

Estructuras de datos: Los árboles se utilizan ampliamente como estructuras de datos para almacenar y organizar información. Los árboles binarios, árboles de búsqueda binaria y árboles B son ejemplos comunes de estructuras de árbol utilizadas para implementar diccionarios, índices y bases de datos.

Algoritmos de búsqueda: Los árboles de búsqueda, como los árboles binarios de búsqueda y los árboles AVL, se utilizan para realizar búsquedas eficientes en grandes conjuntos de datos ordenados. Estos árboles permiten una búsqueda rápida, ya sea para encontrar un valor específico o para realizar búsquedas basadas en rangos.

Compresión de datos: Los árboles de Huffman se utilizan en algoritmos de compresión de datos, como el algoritmo ZIP, para representar de manera eficiente datos con menor cantidad de bits. Estos árboles asignan códigos de bits más cortos a los caracteres más frecuentes y códigos de bits más largos a los caracteres menos frecuentes, reduciendo así el tamaño total del archivo comprimido.

