

Estructura de datos

Juan Pablo Restrepo Uribe

Ing. Biomedico

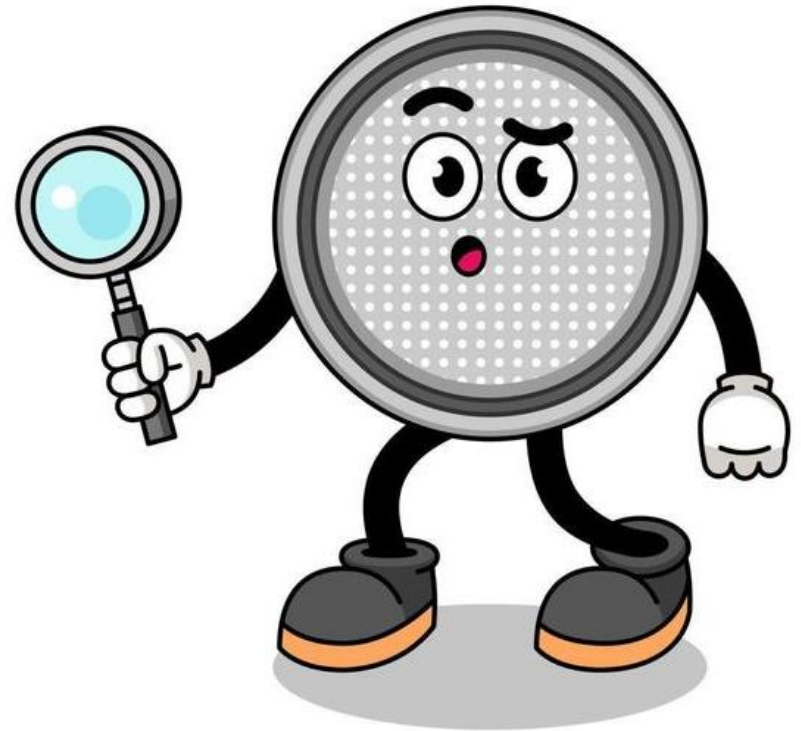
MSc. Automatización y Control Industrial

jprestrepo@correo.iue.edu.co

Institución Universitaria de Envigado

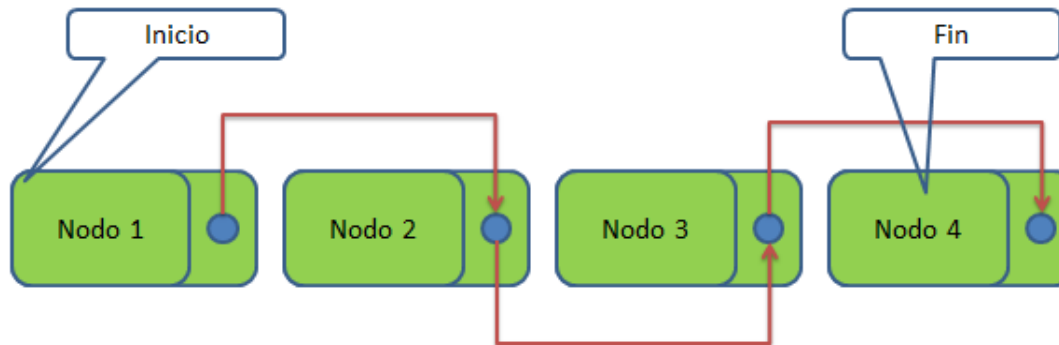
Búsqueda en pilas (Búsqueda lineal)

- Se comienza por el elemento superior de la pila y se compara con el valor que se busca.
- Si el valor no se encuentra, se repite el proceso con el siguiente elemento, y así sucesivamente hasta que se recorra toda la pila o se encuentre el valor.
- La complejidad temporal de este método es $O(n)$, donde n es el número de elementos en la pila.



Búsqueda en pilas (Búsqueda recursiva)

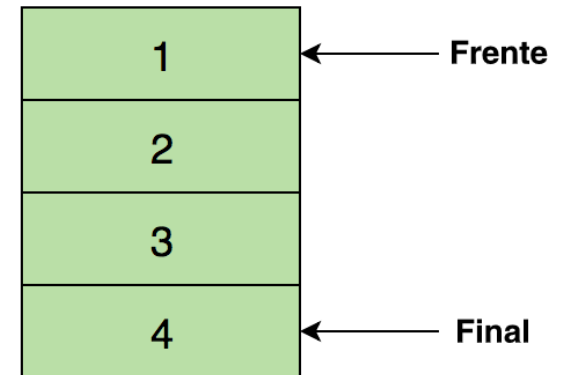
- Se comienza por el elemento superior de la pila y se compara con el valor que se busca.
- Si el valor no se encuentra, se realiza una búsqueda recursiva en el subconjunto de la pila que contiene los elementos que se agregaron antes que el elemento superior.
- La complejidad temporal de este método es $O(n)$, donde n es el número de elementos en la pila.



Limitaciones de la búsqueda en pila

- Las pilas no son una estructura de datos ideal para la búsqueda, ya que la búsqueda lineal tiene una complejidad temporal de $O(n)$.
- Si se necesita realizar una búsqueda en una pila, se recomienda utilizar una estructura de datos más eficiente, como una lista o un árbol.

Eliminar elementos anteriores



Insertar nuevos elementos

Expresiones postfijas

Las expresiones postfijas, también conocidas como notación polaca inversa (NPI), son una forma de escribir expresiones matemáticas donde el operador se coloca después de sus operandos.

La expresión infija $2 + 3$ se escribe como $2\ 3\ +$ en notación postfija.

La expresión infija $(a * b) + c$ se escribe como $a\ b\ *\ c\ +$ en notación postfija.

Expresiones postfijas

- **Ventajas de las expresiones postfijas:**
 - No necesitan paréntesis para indicar el orden de las operaciones, ya que la posición del operador define su precedencia.
 - Son más fáciles de evaluar por computadora, ya que no se requiere un análisis sintáctico complejo.
 - Se utilizan en lenguajes de programación como Forth y PostScript.
- **Desventajas de las expresiones postfijas:**
 - Pueden ser difíciles de leer y escribir para los humanos, ya que no siguen el orden natural de las operaciones matemáticas.
 - Requieren un algoritmo específico para su evaluación.

Expresiones postfijas

Para evaluar una expresión postfija, se utiliza una pila:

- Se recorre la expresión de izquierda a derecha.
- Si se encuentra un operando, se apila en la pila.
- Si se encuentra un operador, se desapilan dos operandos de la pila y se aplica el operador a ellos.
- El resultado de la operación se apila en la pila.
- Se repiten los pasos 2 a 4 hasta que se procesen todos los caracteres de la expresión.
- El valor final que queda en la pila es el resultado de la expresión.

Expresiones postfijas

Por ejemplo, el cálculo: $((1 + 2) * 4) + 3$, se traduciría o notación postfija con la ventaja de no tener que atender a la jerarquía de prioridades:

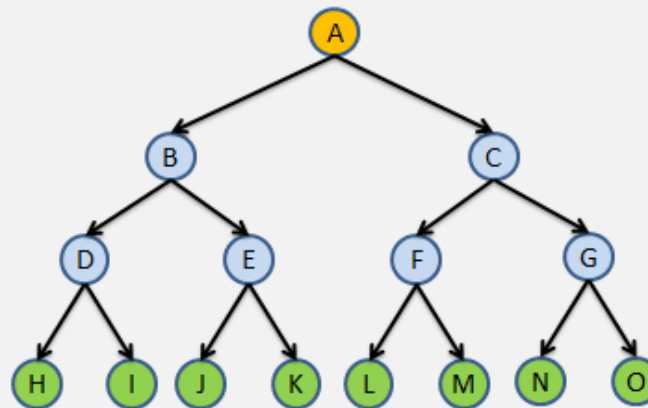
1 2 + 4 * 3 +

ENTRADA	OPERACIÓN	PILA
1	Apilar operando	1
2	Apilar operando	1, 2
+	Sumar operandos*	3
4	Apilar operando	3, 4
*	Multiplicar operandos*	12
3	Apilar operando	12, 3
+	Sumar operandos*	15

El resultado final, 15, se encuentra en la pila, y el puntero apuntado a su dirección. Se lee haciendo pop y la pila vuelve a quedar vacía.

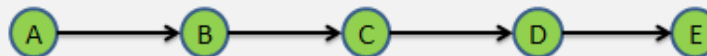
Arboles

Estructura Jerárquica



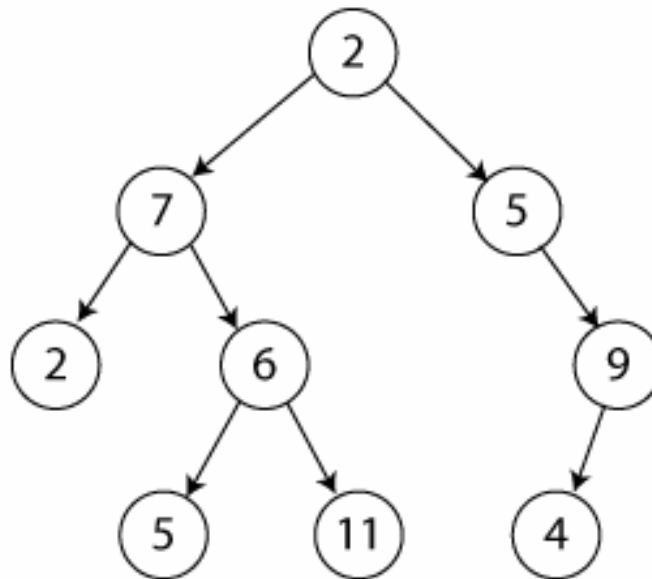
V.S.

Estructura Lineal

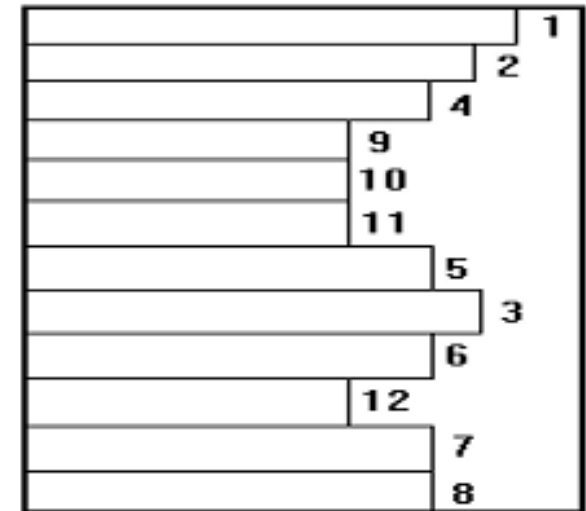
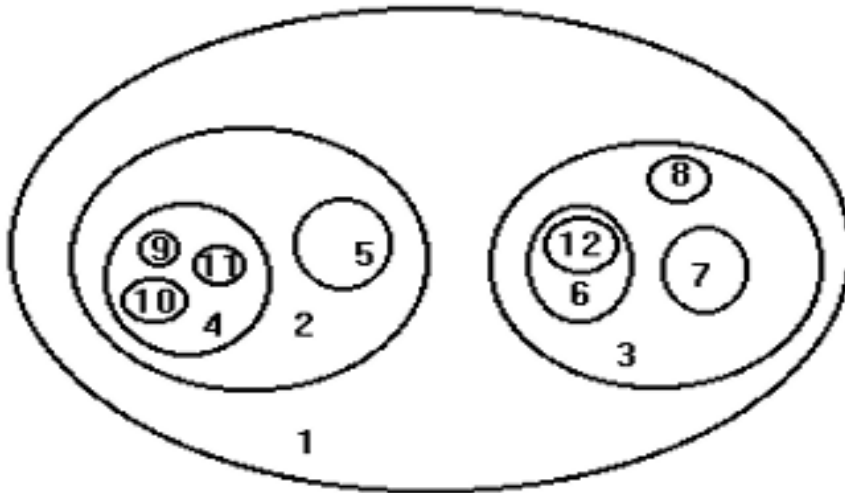


Arboles

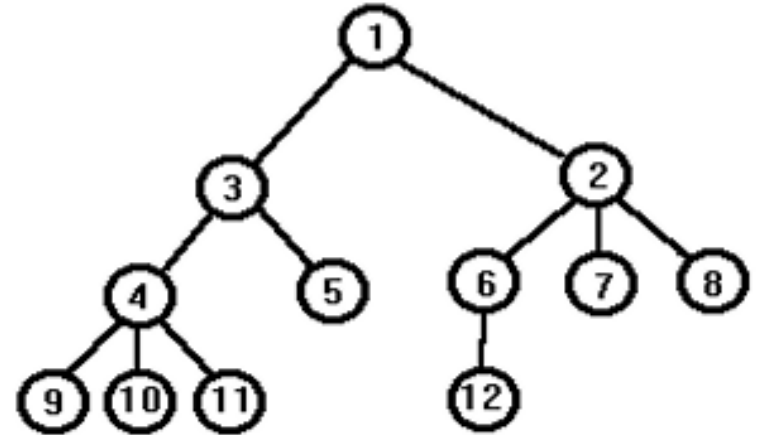
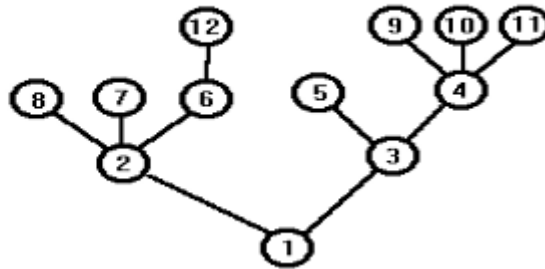
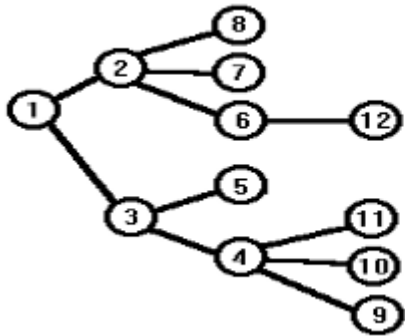
Un árbol es una estructura no lineal formada por un conjunto de nodos y un conjunto de ramas.



Arboles



Arboles

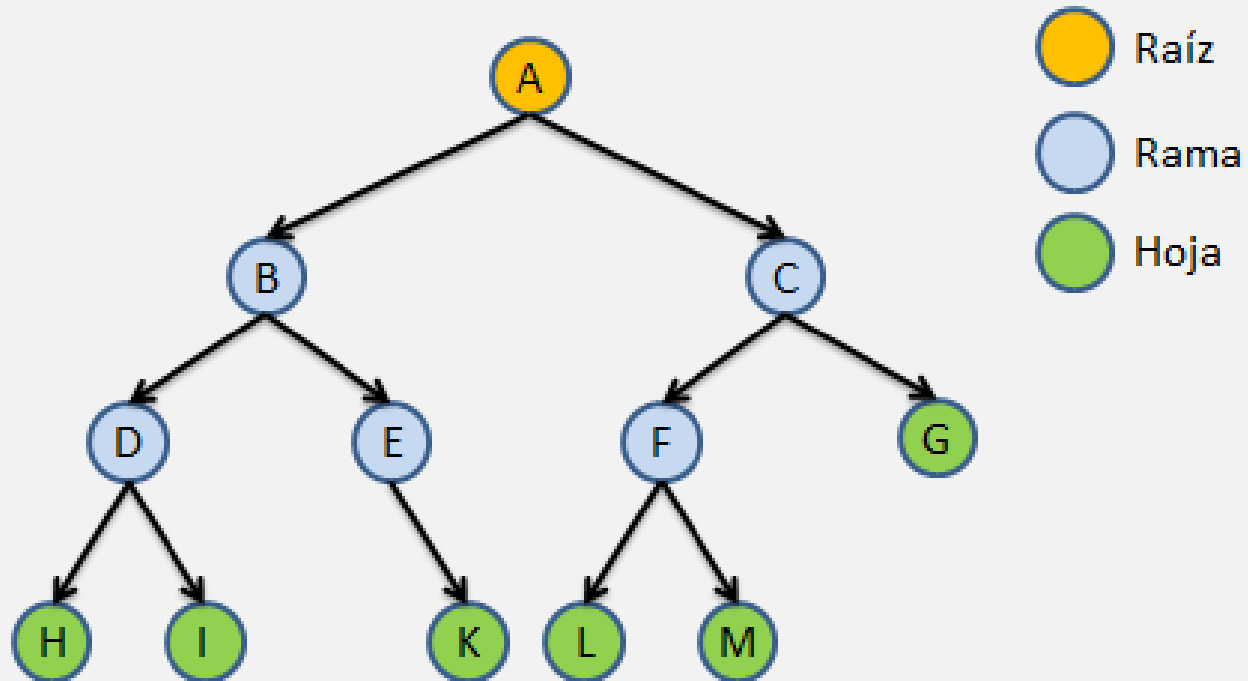


Arboles

- **Nodos:** Se le llama Nodo a cada elemento que contiene un Árbol.
- **Nodo Raíz:** Se refiere al primer nodo de un Árbol, Solo un nodo del Árbol puede ser la Raíz.
- **Nodo Padre:** Se utiliza este término para llamar a todos aquellos nodos que tiene al menos un hijo.
- **Nodo Hijo:** Los hijos son todos aquellos nodos que tiene un padre.
- **Nodo Hermano:** Los nodos hermanos son aquellos nodos que comparte a un mismo padre en común dentro de la estructura.
- **Nodo Hoja:** Son todos aquellos nodos que no tienen hijos, los cuales siempre se encuentran en los extremos de la estructura.
- **Nodo Rama:** Estos son todos aquellos nodos que no son la raíz y que además tiene al menos un hijo.

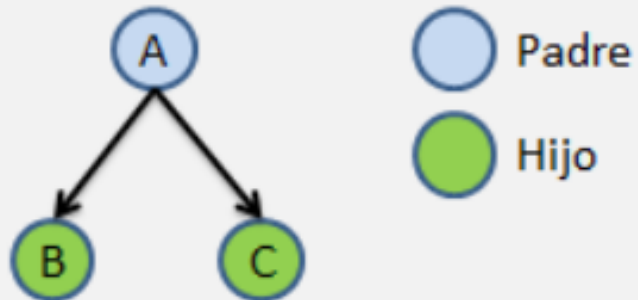
Arboles

Tipos de nodos en un Árbol

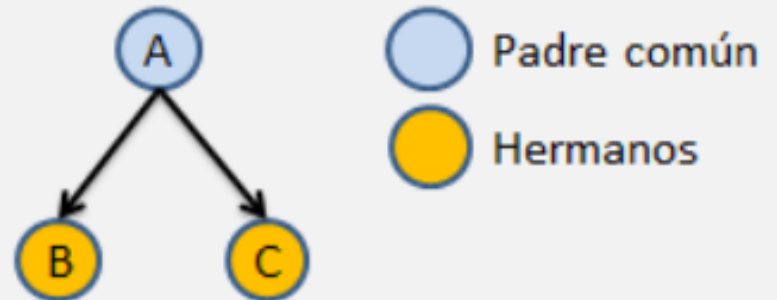


Arboles

Nodos Padre e Hijos



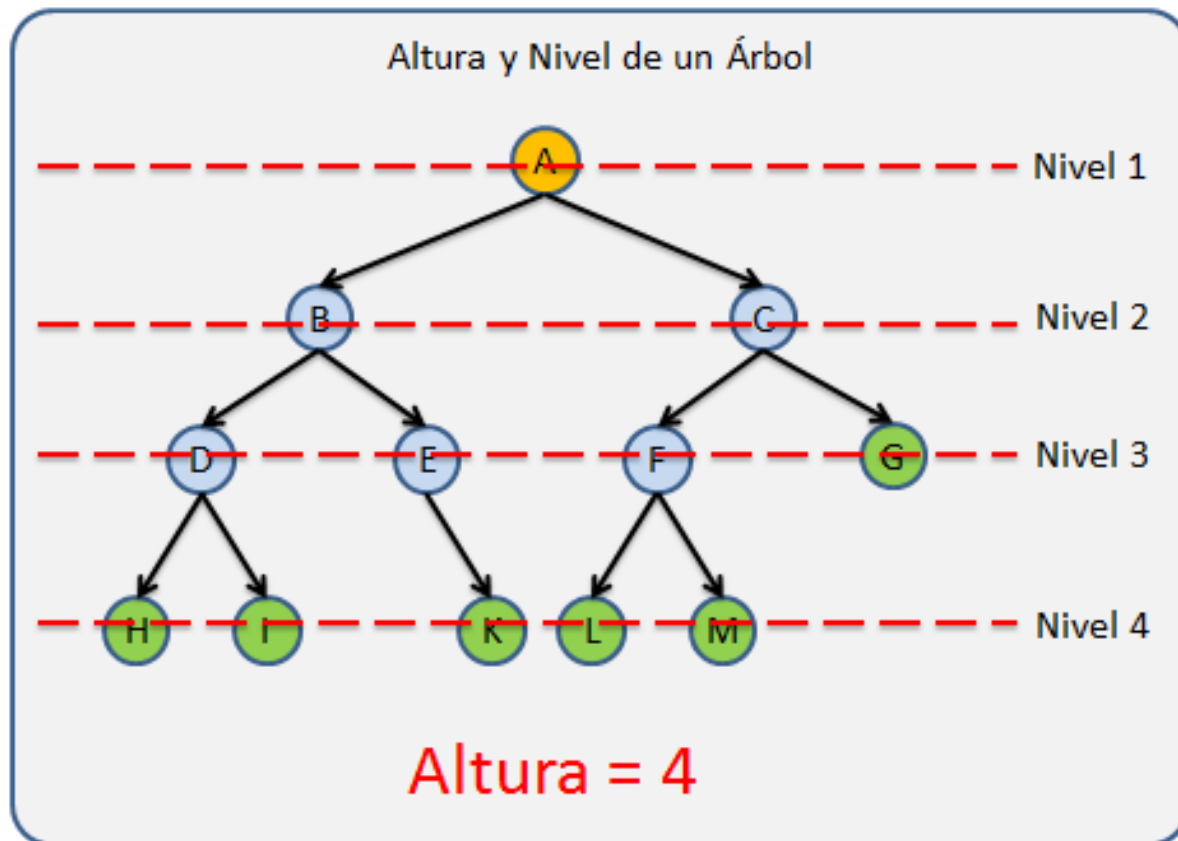
Nodos Padre e Hijos



Arboles

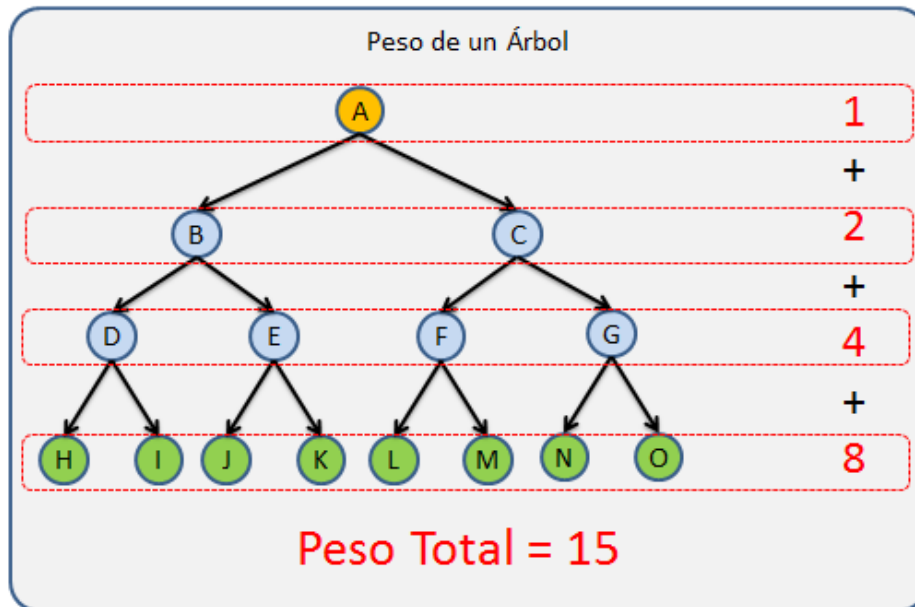
- **Nivel:** Nos referimos como nivel a cada generación dentro del árbol. Por ejemplo, cuando a un nodo hoja le agregamos un hijo, el nodo hoja pasa a ser un nodo rama, pero además el árbol crece una generación por lo que el Árbol tiene un nivel más. Cada generación tiene un número de Nivel distinto que las demás generaciones.
 - Un árbol vacío tiene 0 niveles
 - El nivel de la Raíz es 1
 - El nivel de cada nodo se calculado contando cuantos nodos existen sobre él, hasta llegar a la raíz + 1, y de forma inversa también se podría, contar cuantos nodos existes desde la raíz hasta el nodo buscado + 1.
- **Altura:** Le llamamos Altura al número máximo de niveles de un Árbol.

Arboles



Arboles

Peso: Conocemos como peso a el número de nodos que tiene un Árbol. Este factor es importante porque nos da una idea del tamaño del árbol y el tamaño en memoria que nos puede ocupar en tiempo de ejecución (**Complejidad Espacial en análisis de algoritmos.**)



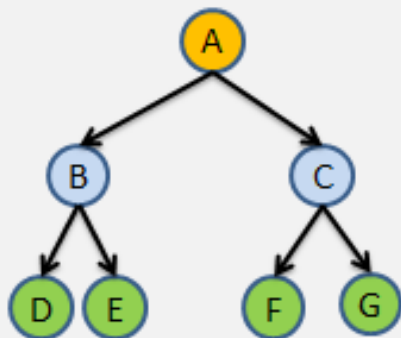
El peso se puede calcular mediante cualquier tipo de recorrido el cual valla contando los nodos a medida que avanza sobre la estructura.

Arboles

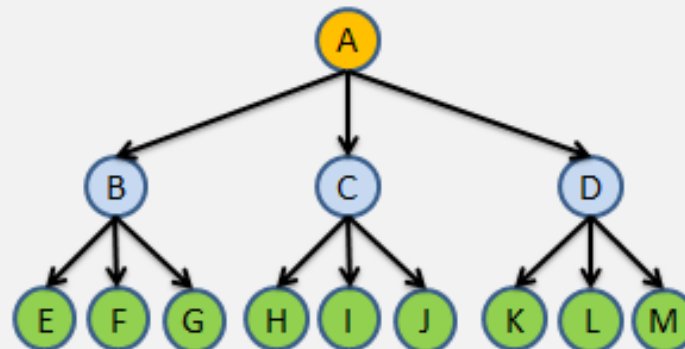
Orden: El Orden de un árbol es el número máximo de hijos que puede tener un Nodo.

- Árbol con Orden = 1 no tendría sentido ya que sería una estructura lineal. ya que cada nodo solo podría tener un Hijo.

Orden de un Árbol



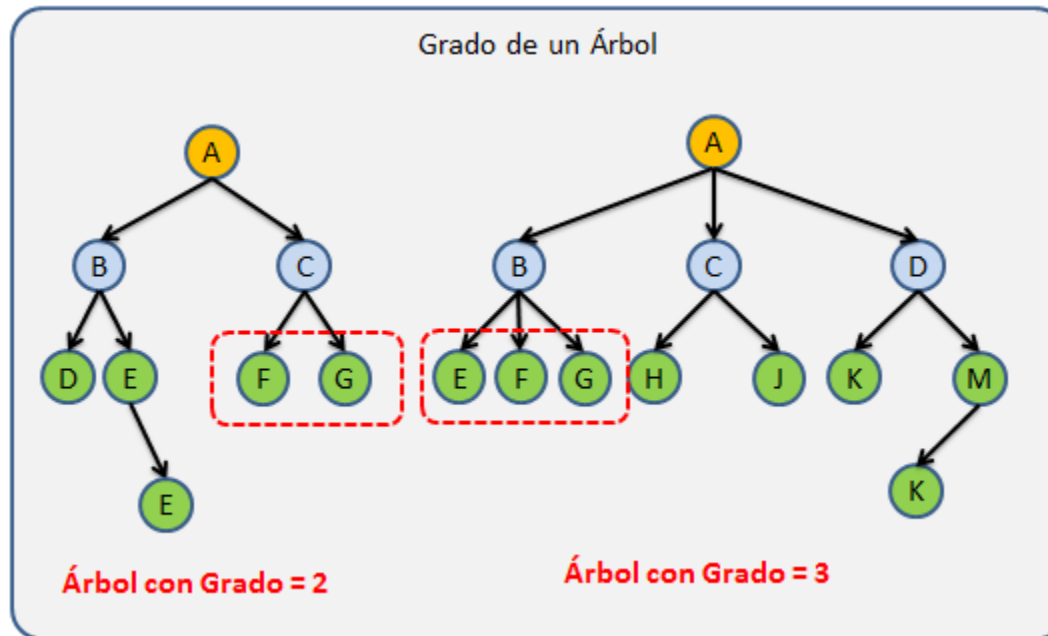
Árbol con Orden = 2



Árbol con Orden = 3

Arboles

Grado: El grado se refiere al número mayor de hijos que tiene alguno de los nodos del Árbol y está limitado por el Orden, ya que este indica el número máximo de hijos que puede tener un nodo.

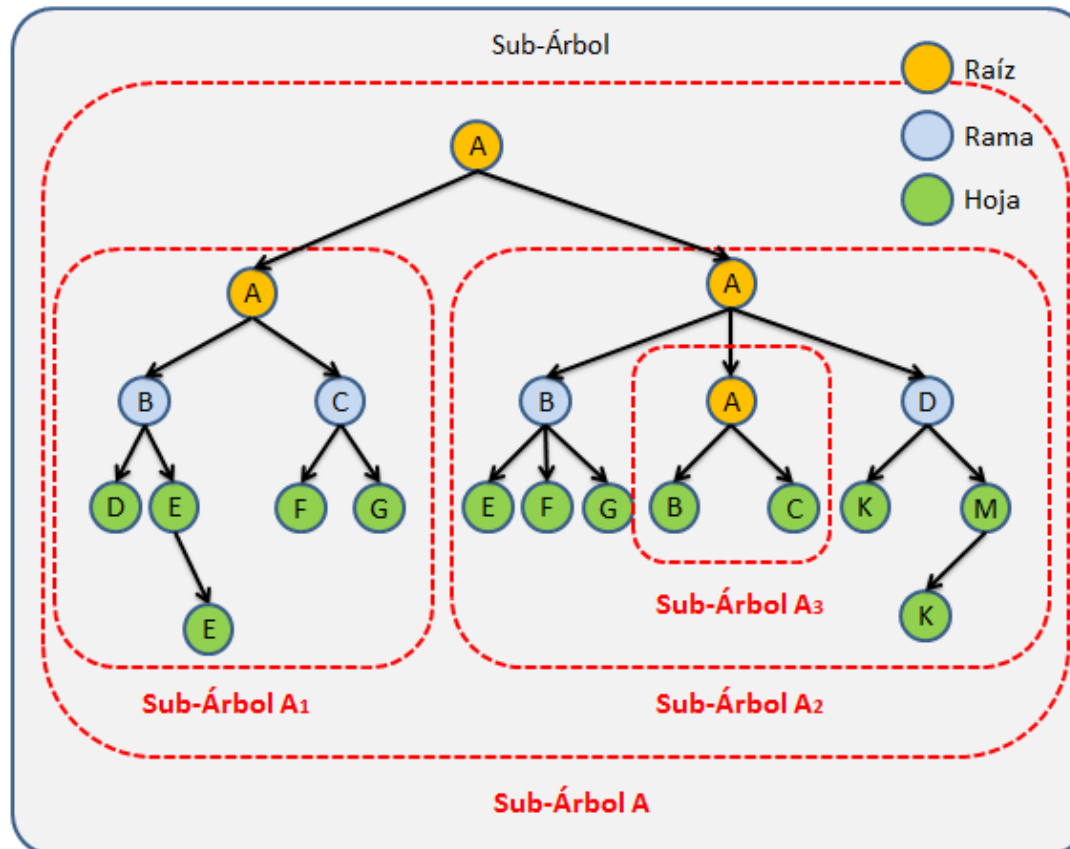


Arboles

Sub-Árbol: Conocemos como Sub-Árbol a todo Árbol generado a partir de una sección determinada del Árbol, por lo que podemos decir que un Árbol es un nodo Raíz con N Sub-Árboles.

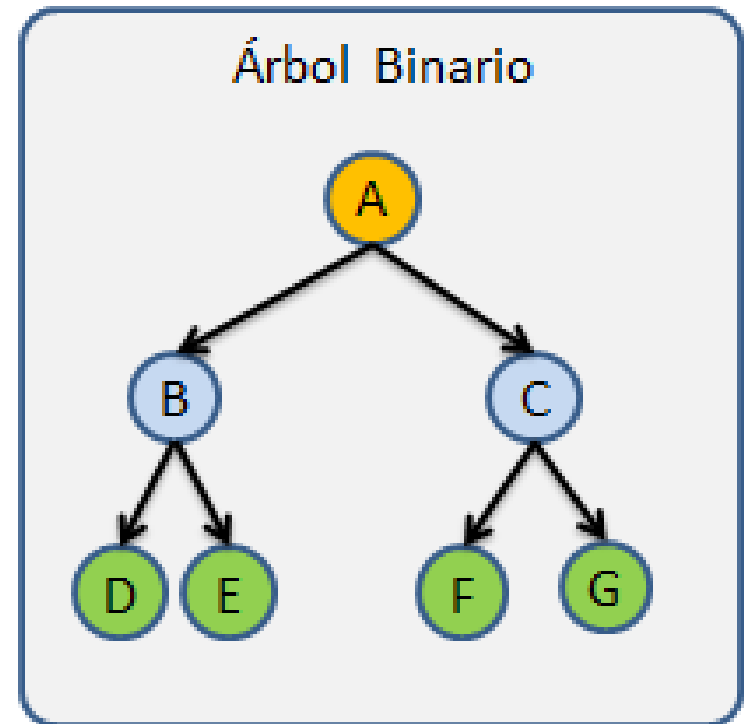
Existen escenarios donde podemos sacar un Sub-Árboles del Árbol para procesarlo de forma separada, de esta forma el Sub-Árboles pasa a ser un Árbol independiente, también podemos eliminar Sub-Árboles completos, Agregarlos, entre otras operaciones.

Arboles



Arboles

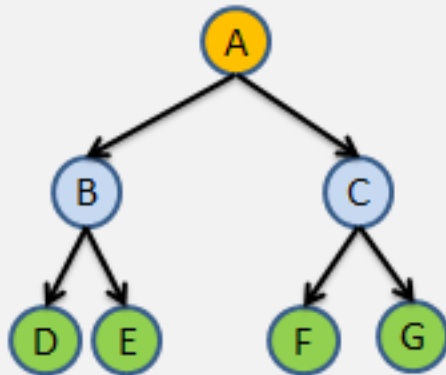
- **Árbol n-ario:** los árboles n-arios son aquellos arboles donde el número máximo de hijos por nodo es de N, en la figura 7 podemos apreciar dos árboles con grado 2 y grado 3, estos dos árboles también los podemos definir como Árbol n-ario con $n = 2$ y $n=3$ respectivamente.
- **Árboles binarios:** Esta estructura se caracteriza por que cada nodo solo puede tener máximo 2 hijos, dicho de otra manera, es un Árbol n-ario de Grado 2.



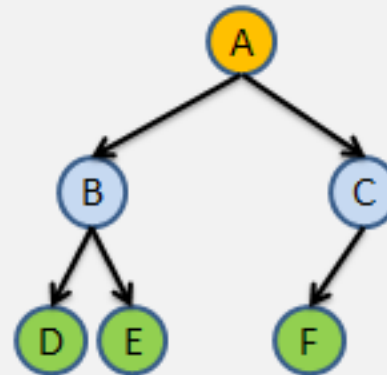
Arboles

- **Árbol binario lleno:** Es aquel que el que todos los nodos tienen cero o 2 hijos con excepción de la Raíz.

Árbol Binario Lleno

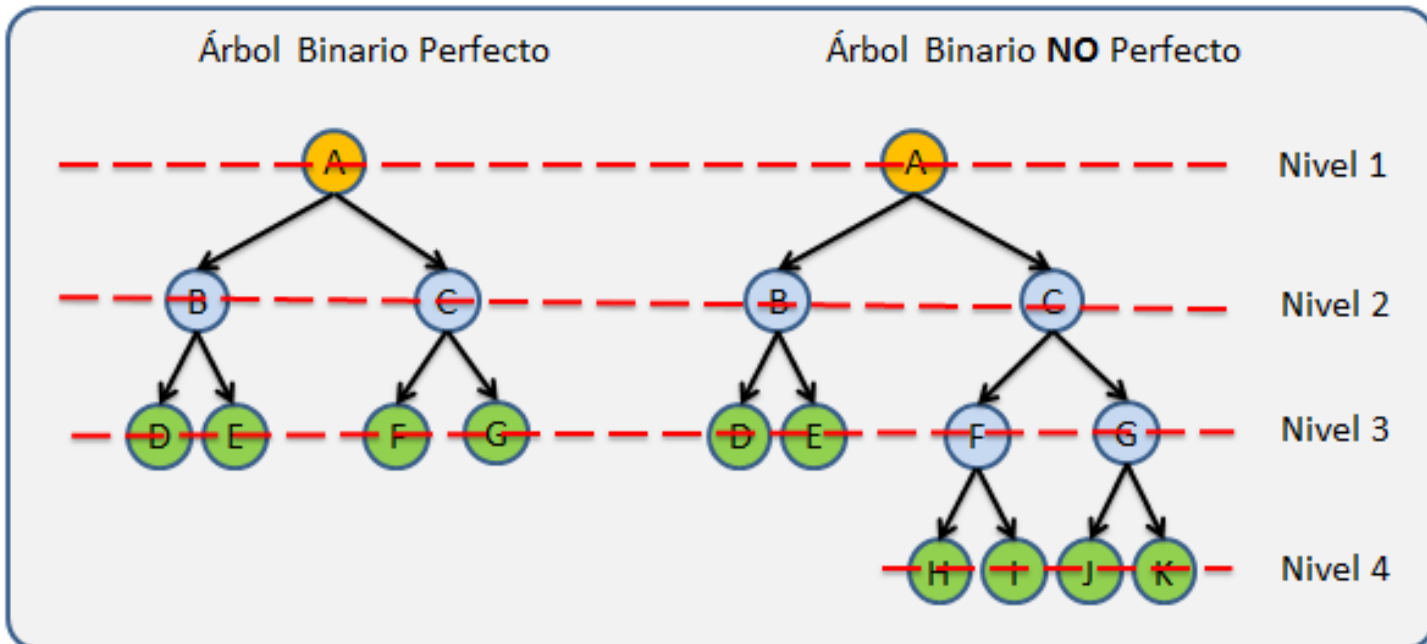


Árbol Binario **NO** Lleno

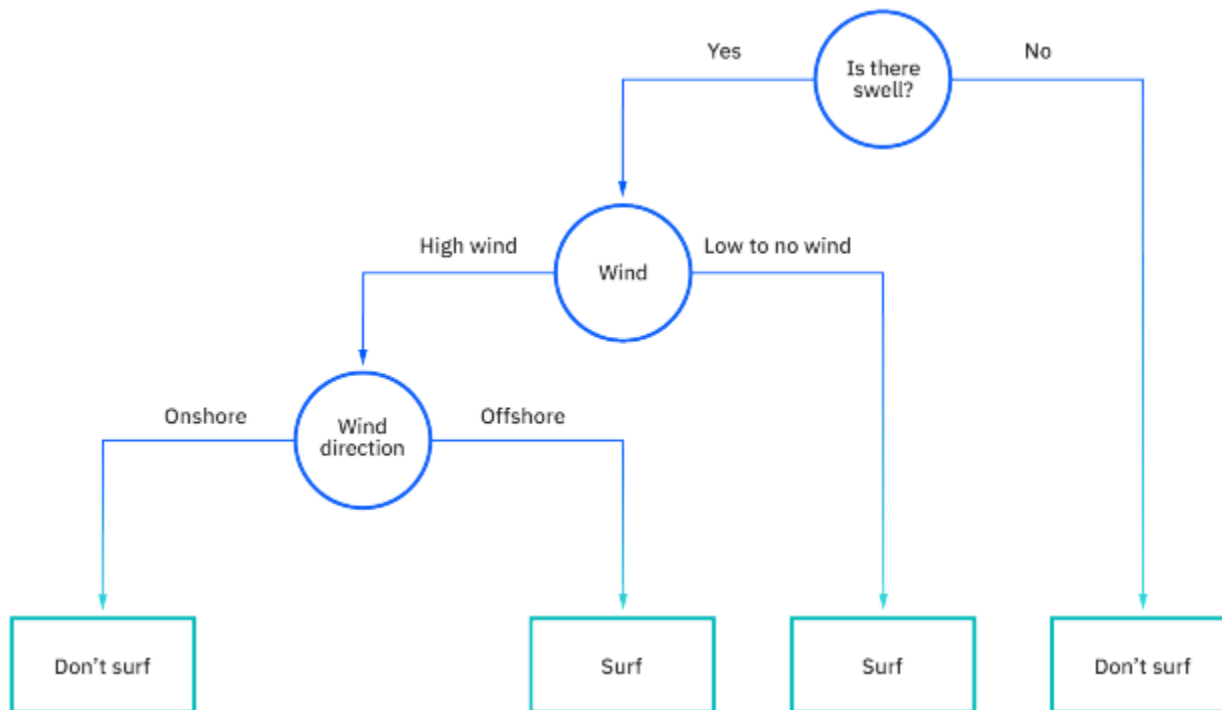


Arboles

- **Árbol binario perfecto:** Es un Árbol lleno en donde todas las Hojas están en el mismo Nivel.



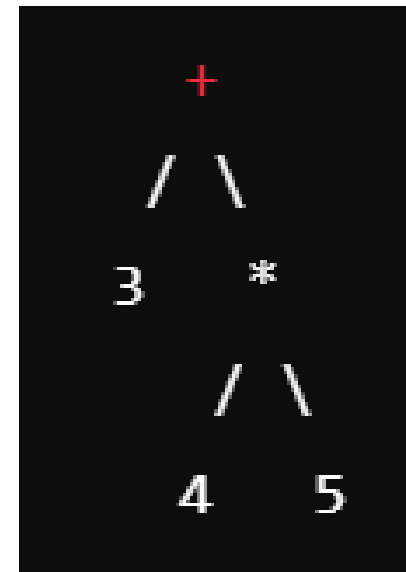
Arboles



Arboles

Árboles binarios en expresiones matemáticas:

Un compilador puede usar un árbol binario para representar expresiones matemáticas o booleanas. Cada nodo del árbol contiene un operador, mientras que los nodos hoja contienen los operandos. Por ejemplo, la expresión $3 + (4 * 5)$ se representaría como:

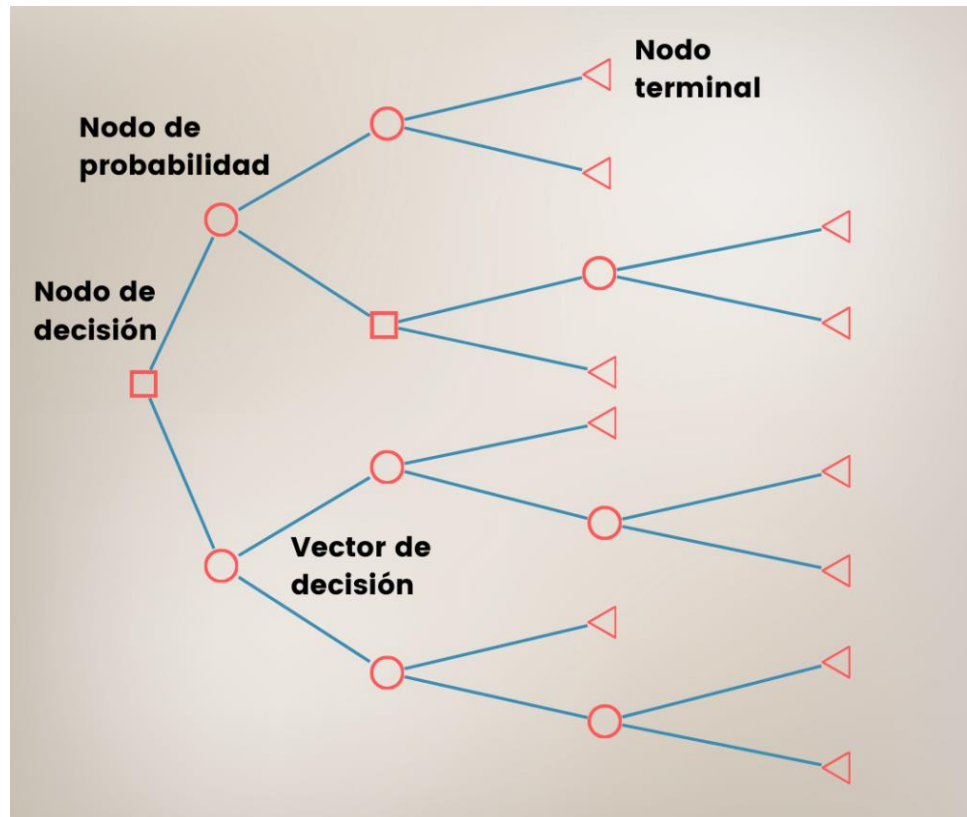


Arboles

Árboles de decisión en Machine Learning:

En Machine Learning, los árboles de decisión se utilizan para tomar decisiones basadas en condiciones. Cada nodo representa una decisión (o división de los datos), y cada rama conduce a un resultado. Por ejemplo, un árbol de decisión puede usarse para predecir si un cliente comprará o no un producto en función de su historial de compras y otros datos.

Arboles



Arboles

Arboles Binarios en Motores de Juegos (Octrees y Quadrees)

En gráficos 3D, se utilizan octrees para subdividir el espacio en cubos más pequeños y organizar los objetos en el entorno de manera eficiente. Esto permite realizar cálculos de colisión y renderización de manera más rápida. En gráficos 2D, se utilizan quadrees para dividir el espacio en cuadrantes.