Título: Ubicación de un Nodo por su Representación Binaria
Autor: Luis R. Morera González

En este artículo introduciremos un algoritmo de carácter netamente geométrico para ubicar en un árbol natural la representación binaria encontrada utilizando los algoritmos de las operaciones básicas (suma, resta, multiplicación y división) descritas en artículos anteriores.

Sea n, m dos nodos en un árbol natural, el siguiente algoritmo permite ubicar la suma, resta, multiplicación y división de estos nodos, ha partir de la representación binaria encontrada utilizando cualquiera de los algoritmo utilizados.

Algoritmo para la ubicación de un nodo por su representación binaria

Sean n, m dos nodos en un árbol natural y b = $\{b_X, \ldots., b_1, b_0\}$ la representación binaria de la de las operaciones basicas.

Paso 1: $P_0 \leftarrow 1$ “posición en el árbol natura”

Paso 2: Si hay un solo bit encendido en la representación binaria pasar al **Paso 5**.

Paso 3: Sea DL(i,j) = i – j la diferencia en niveles entre los primeros dos bit encendidos de la representación binaria “b”, estamos suponiendo que i > j son los enteros más grandes para los cuales b_i=1, b_j=1.

a) $b_i \leftarrow 0$
b) $P_1 \leftarrow 2^{DL(i,j)}P_0 + 1$
c) $P_0 \leftarrow P_1$

Paso 4: Regresar al **Paso 2**.

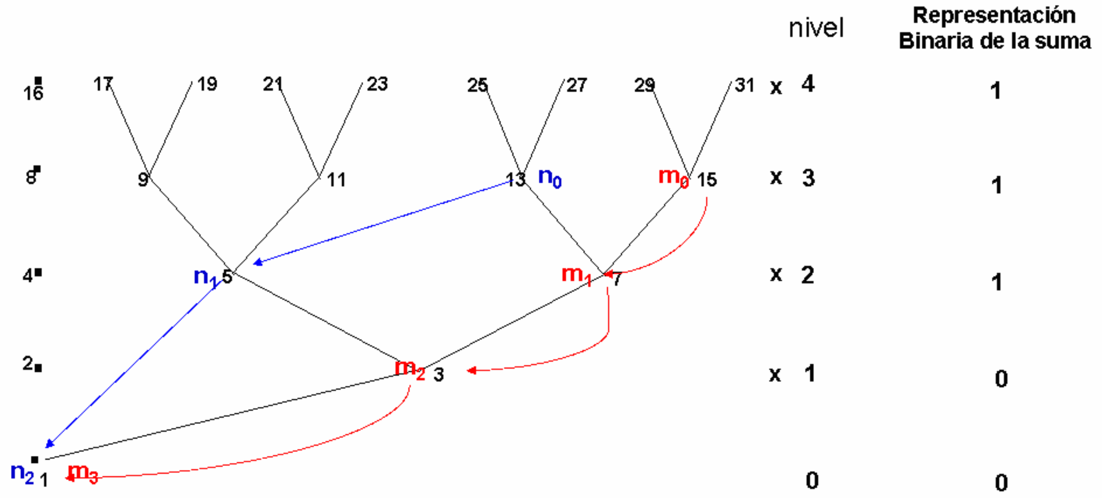
Paso 5: La suma esta dada por $P_0 \leftarrow 2^i P_0$ ╣

Gráficamente cada vez que se completa el **Paso 3**, pasamos al nodo cuya etiqueta es P_0. Al Final del **Paso 5,** P_0 es la etiqueta de la suma.

Inicialmente explicaré el algoritmo anterior haciendo todos los pasos, para encontrar la posición del nodo que representa la suma de n = 13 y m = 15.

Para esto inicialmente buscamos la representación binaria de la suma utilizando el algoritmo de la suma, como se muestra en la **Figura 1**.

Figura 1

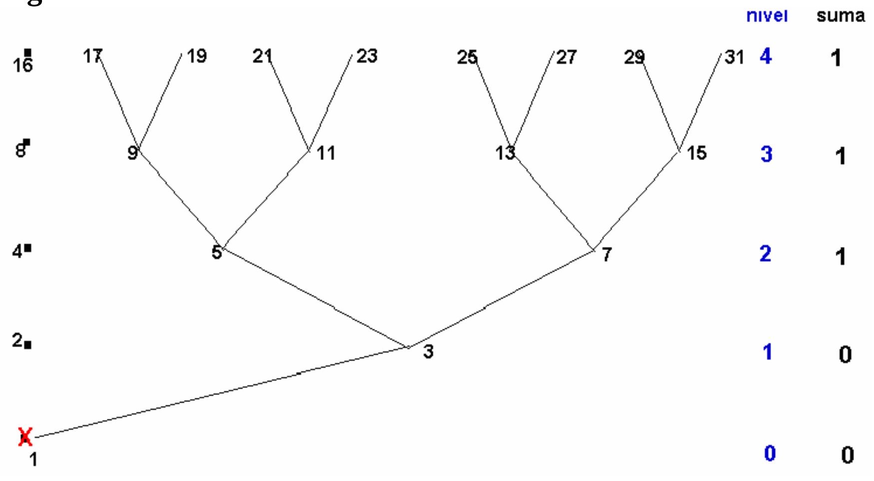


Esto implica que b = {b_4=1, b_3=1, b_2=1, b_1=0, b_0=0}.

Utilizando el algoritmo anterior tenemos: "recuerde que ya las orbitas de n y m, no son necesarias"

Paso 1: $P_0 \leftarrow 1$ "esto es que nos ubicamos en la raíz", como se muestra en la **Figura 2**.

Figura 2



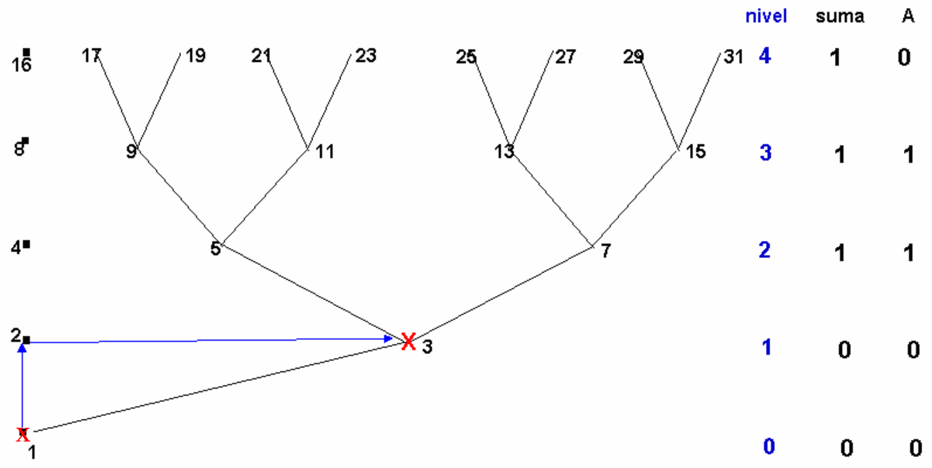
Paso 2: Como no hay un solo bit encendido en la representación binaria continúe.

Paso 3: Como DL(4,3) = 4 – 3 = 1

a) $b_4 \leftarrow 0$
b) $P_1 \leftarrow 2^1 \times 1 + 1 = 3$
c) $P_0 \leftarrow P_1$

Esto es nos posicionamos en el nodo etiquetado con 3, este es el nuevo P_0 "gráficamente se sube un nivel y se mueve a la derecha la posición x anterior" y creamos la tabla A en la cual se puso 0 en b_4. Como se muestra en la **Figura 3**.

Figura 3



Paso 4: Regresar al **Paso 2** utilizando la columna A de la figura anterior. Esto es b = {b_4=0, b_3=1, b_2=1, b_1=0, b_0=0}.

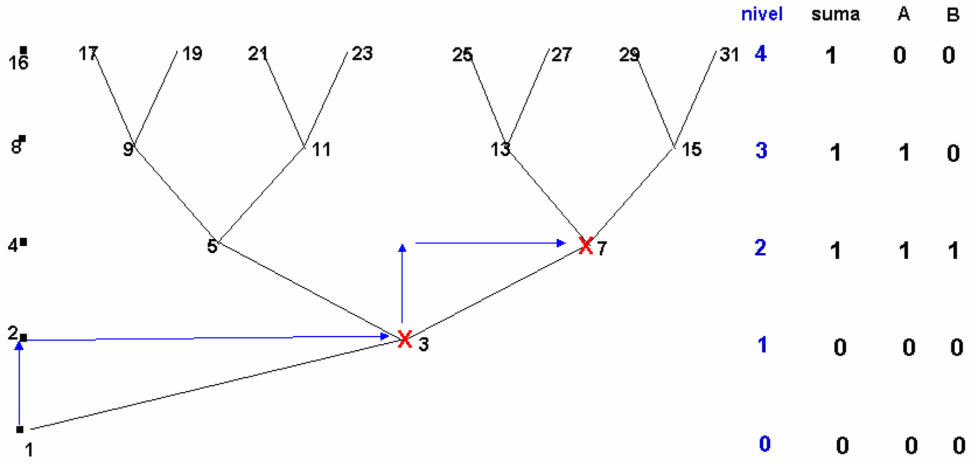
Paso 2: Como no hay un solo bit encendido en la representación binaria continúe.

Paso 3: Como DL(3,2) = 3 – 2 = 1

a) $b_3 \leftarrow 0$
b) $P_1 \leftarrow 2^1 \times 3 + 1 = 7$
c) $P_0 \leftarrow P_1$

Esto es nos posicionamos en el nodo etiquetado con 7, este es el nuevo P_0 "gráficamente se sube un nivel y se mueve a la derecha la posición x anterior" y creamos la tabla B en la cual se puso 0 en b_3. Como se muestra en la **Figura 4**.

Figura 4

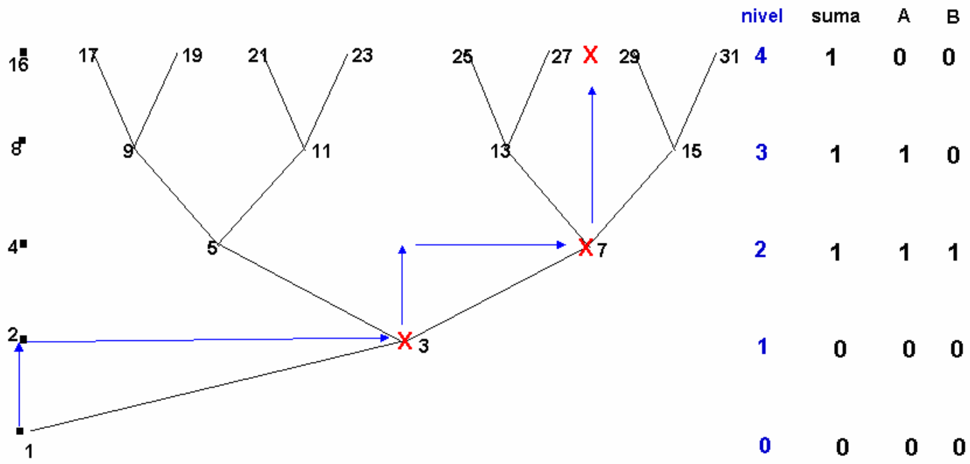


Paso 4: Regresar al **Paso 2** utilizando la columna B de la figura anterior. Esto es b = {b_4=0, b_3=0, b_2=1, b_1=0, b_0=0}.

Paso 2: Como hay un solo bit encendido en la representación binaria pasar al **Paso 5**.

Paso 5: La suma esta dada por $P_0 \leftarrow 2^2 \times 7 = 28$. Esto es subir 2 niveles P_0 "esto es así debido a que el bit encendido en la columna B de la figura anterior esta en el nivel 2". Como se muestra el la **Figura 5**.

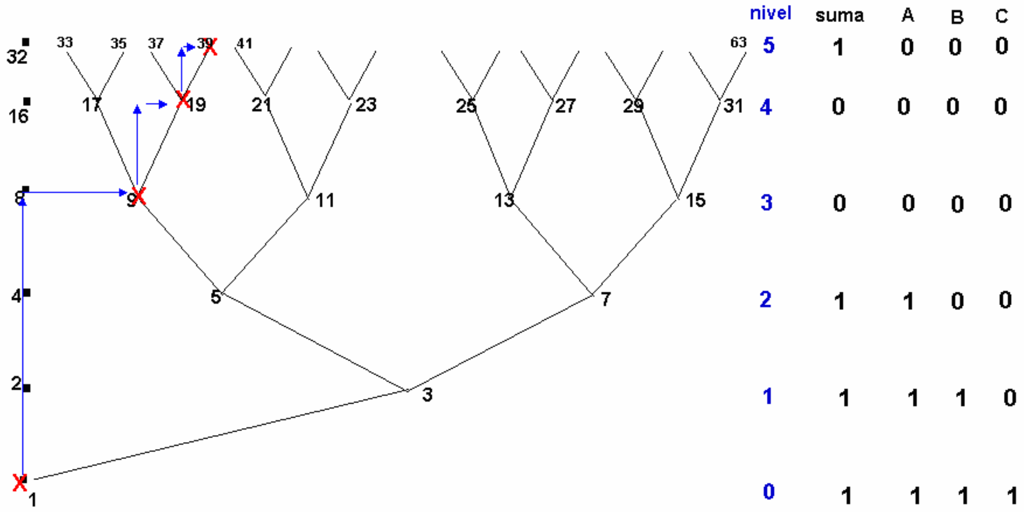
Figura 5



Note que la x en el nivel 4 de la figura anterior representa el número 28 que es el decimal que representa la representación binaria 11100_2.

La **Figura 6** muestra la posición en el árbol natural de la suma de 9 + 30, utilizando la representación binaria de la suma y el algoritmo estudiado anteriormente.

Figura 6



La ubicación esta dada por la x en el nivel 5. Este nodo esta etiquetado con el número 39.

Justificación del algoritmo para la ubicación de un nodo por su representación binaria

Sea b = {$b_{Max\{L(n),L(m)\}+1}$, …., b_1, b_0} la representación binaria de la suma de n y m, donde

$$b_i = \begin{cases} 1 & \text{si } 2^i \text{ esta en la representación binaria de la suma} \\ 0 & \text{en caso contrario} \end{cases}$$

Sea $b_{i_1} = b_{i_2} = ... = b_{i_{k-1}} = b_{i_k} = 1$ donde $i_1 > i_2 > .... > i_{k-1} > i_k$ por lo cual tenemos que la suma queda representada por $S = 2^{i_1} + 2^{i_2} + ... + 2^{i_{k-1}} + 2^{i_k}$. Para evaluar esta suma se usa el equivalente gráfico de la regla de Hörner.

Utilizando el algoritmo y la suma “S”, tenemos:

$P_0 = 1$

Sacando factor común entre los primeros dos términos de S

$$S = 2^{i_2}(2^{i_1 - i_2} + 1) + 2^{i_3} + ... + 2^{i_{k-1}} + 2^{i_k} = 2^{i_2} P_1 + 2^{i_3} + ... + 2^{i_{k-1}} + 2^{i_k}$$

Ahora $P_0 \leftarrow P_1$ esto implica: (**1**) $S = 2^{i_2} P_0 + 2^{i_3} + ... + 2^{i_{k-1}} + 2^{i_k}$

Sacando factor común entre los primeros dos términos de (**1**)

$$S = 2^{i_3}(2^{i_2 - i_3} P_0 + 1) + 2^{i_4} + ... + 2^{i_{k-1}} + 2^{i_k} = 2^{i_3} P_1 + 2^{i_4} + ... + 2^{i_{k-1}} + 2^{i_k}$$

Ahora $P_0 \leftarrow P_1$ esto implica: (**2**) $S = 2^{i_3} P_0 + 2^{i_4} + ... + 2^{i_{k-1}} + 2^{i_k}$

Si continuamos con el proceso tenemos:

$S = 2^{i_k}(2^{i_{k-1}-i_k} P_0 + 1) = 2^{i_k} P_1$

Por último $P_0 \leftarrow P_1$ esto implica: $S = 2^{i_k} P_0$

Por lo tanto: $P_0 = 2^{i_k} P_0$ ╣