

## Algoritmo de Huffman

### Caso Práctico

Dado un archivo que contiene la siguiente frase:

**“TOMAMOS\_COMO\_EJEMPLO\_ESTA\_FRASE\_“**

( \_ ) El guion bajo representa los blancos.

Se procederá a implementar el algoritmo de Huffman, generando la tabla de frecuencias y el árbol binario de búsqueda. Luego de ello se compactará dicho archivo.

La implementación del mismo consta de cuatro etapas,

- Generación de la tabla de Frecuencias
- Generación del Arbol Binario
- Compresión o compactación de datos.
- Descompresión o descompactación de datos.

las dos primeras etapas hacen a la construcción de las estructuras de memoria, y las dos últimas al tratamiento de los datos.

### **Generación de la tabla de frecuencias**

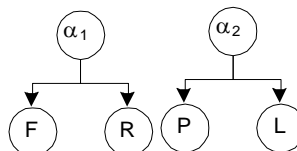
Se realiza un proceso de conteo de los caracteres del archivo y se van cargando en una estructura de memoria que contiene a la tabla de frecuencias (puede ser una lista linkeada en la cual cada nodo cuenta con los campos necesarios de la tabla).

Para facilitar la explicación, procedimos a ordenar los elementos de la tabla de frecuencia de mayor a menor.

<b>Status</b>	<b>Char</b>	<b>Freq</b>	<b>Code</b>	<b>Dir_tree</b>
	Blanco	5		
	O	5		
	M	4		
	E	4		
	A	3		
	S	3		
	T	2		
	C	1		
	J	1		
	P	1		
	L	1		
	F	1		
	R	1		

La cantidad total de caracteres es  $32 \times 8 \text{ bits} = 256 \text{ bits}$



[illegible]



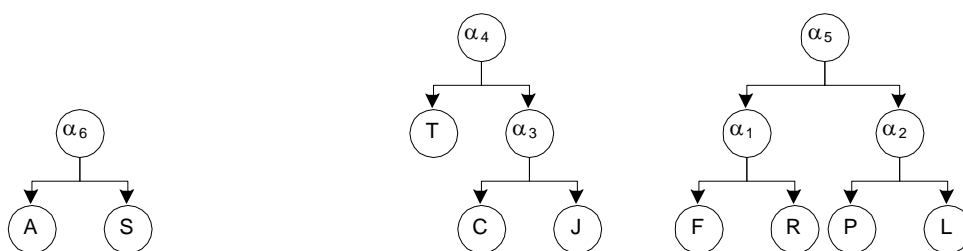




Paso 6

Tomo los próximos dos caracteres de menor frecuencia, A y S. Armo un sub-árbol con raíz " $\alpha_6$ " en memoria con ambos nodos A como hijo izq. y S como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_6$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones de los tres nodos y el status de A y S.

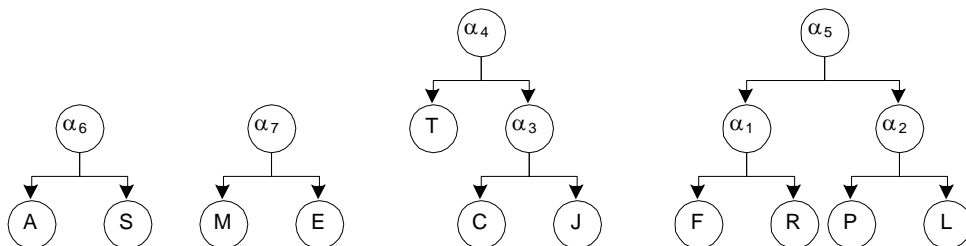
Status	Char	Freq	Code	Dir_tree
	Blanco	5		
	O	5		
	M	4		
	E	4		
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
	$\alpha_4$	4		dir ( $\alpha_4$ )
	$\alpha_5$	4		dir ( $\alpha_5$ )
	$\alpha_6$	6		dir ( $\alpha_6$ )



Paso 7

Tomo los próximos dos caracteres de menor frecuencia, M y E. Armo un sub-árbol con raíz " $\alpha_7$ " en memoria con ambos nodos M como hijo izq. y E como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_7$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones de los tres nodos y el status de M y E.

Status	Char	Freq	Code	Dir_tree
	Blanco	5		
	O	5		
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
	$\alpha_4$	4		dir ( $\alpha_4$ )
	$\alpha_5$	4		dir ( $\alpha_5$ )
	$\alpha_6$	6		dir ( $\alpha_6$ )
	$\alpha_7$	8		dir ( $\alpha_7$ )

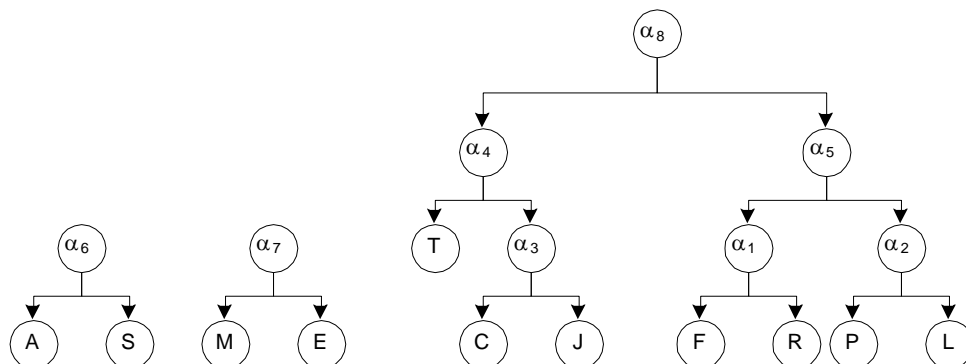




Paso 8

Tomo los próximos dos caracteres de menor frecuencia,  $\alpha_4$  y  $\alpha_5$ . Armo un sub-árbol con raíz " $\alpha_8$ " en memoria con ambos nodos  $\alpha_4$  como hijo izq. y  $\alpha_5$  como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_8$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones del nodo raíz y el status de  $\alpha_4$  y  $\alpha_5$ .

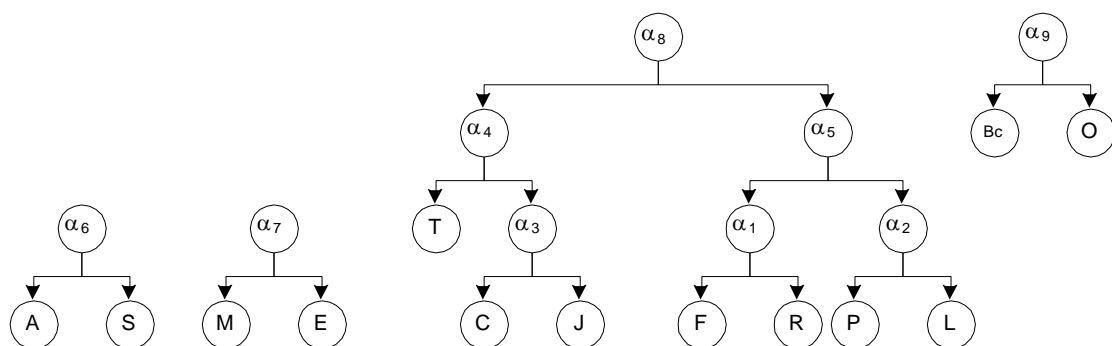
Status	Char	Freq	Code	Dir_tree
	Blanco	5		
	O	5		
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
<b>x</b>	$\alpha_4$	4		dir ( $\alpha_4$ )
<b>x</b>	$\alpha_5$	4		dir ( $\alpha_5$ )
	$\alpha_6$	6		dir ( $\alpha_6$ )
	$\alpha_7$	8		dir ( $\alpha_7$ )
	$\alpha_8$	<b>8</b>		<b>dir (<math>\alpha_8</math>)</b>



Paso 9

Tomo los próximos dos caracteres de menor frecuencia, Blanco y O. Armo un sub-árbol con raíz " $\alpha_9$ " en memoria con ambos nodos Blanco como hijo izq. y O como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_9$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones de los tres nodos y el status de Blanco y O.

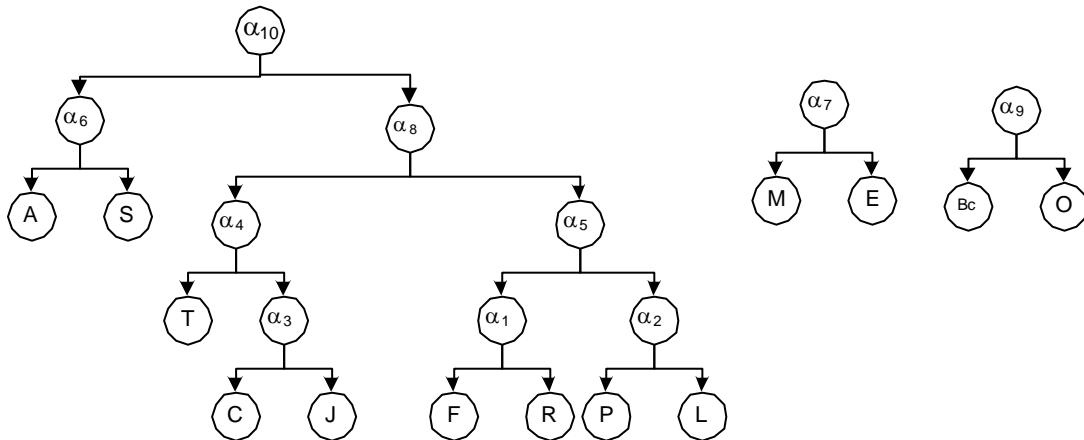
Status	Char	Freq	Code	Dir_tree
x	Blanco	5		dir (Bco)
x	O	5		dir (O)
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
x	$\alpha_4$	4		dir ( $\alpha_4$ )
x	$\alpha_5$	4		dir ( $\alpha_5$ )
	$\alpha_6$	6		dir ( $\alpha_6$ )
	$\alpha_7$	8		dir ( $\alpha_7$ )
	$\alpha_8$	8		dir ( $\alpha_8$ )
	$\alpha_9$	10		dir ( $\alpha_9$ )



Paso 10

Tomo los próximos dos caracteres de menor frecuencia,  $\alpha_6$  y  $\alpha_7$ . Armo un sub-árbol con raíz " $\alpha_{10}$ " en memoria con ambos nodos  $\alpha_6$  como hijo izq. y  $\alpha_7$  como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_{10}$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones del nodo raíz y el status de  $\alpha_6$  y  $\alpha_7$ .

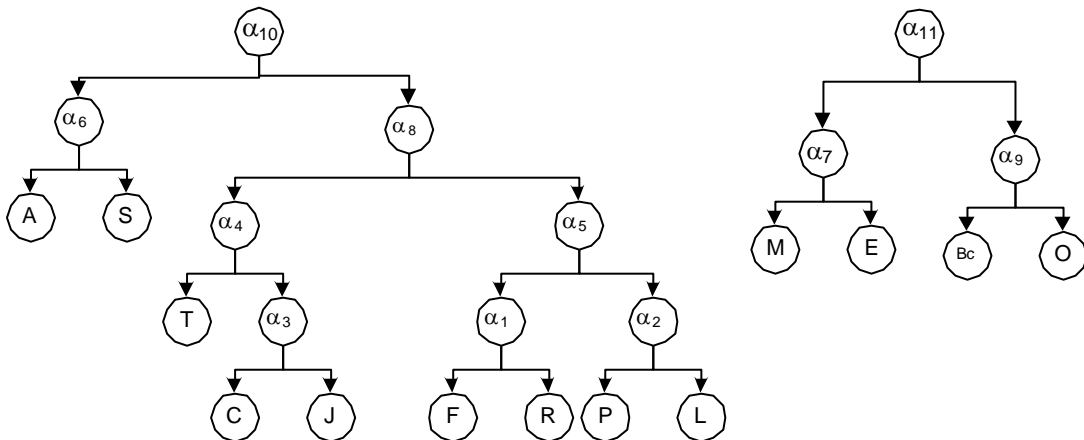
Status	Char	Freq	Code	Dir_tree
x	Blanco	5		dir (Bco)
x	O	5		dir (O)
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
x	$\alpha_4$	4		dir ( $\alpha_4$ )
x	$\alpha_5$	4		dir ( $\alpha_5$ )
<b>x</b>	$\alpha_6$	6		dir ( $\alpha_6$ )
	$\alpha_7$	8		dir ( $\alpha_7$ )
<b>x</b>	$\alpha_8$	8		dir ( $\alpha_8$ )
	$\alpha_9$	10		dir ( $\alpha_9$ )
	$\alpha_{10}$	14		dir ( $\alpha_{10}$ )



Paso 11

Tomo los próximos dos caracteres de menor frecuencia,  $\alpha_8$  y  $\alpha_9$ . Armo un sub-árbol con raíz " $\alpha_{11}$ " en memoria con ambos nodos  $\alpha_8$  como hijo izq. y  $\alpha_9$  como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_{11}$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones del nodo raíz y el status de  $\alpha_8$  y  $\alpha_9$ .

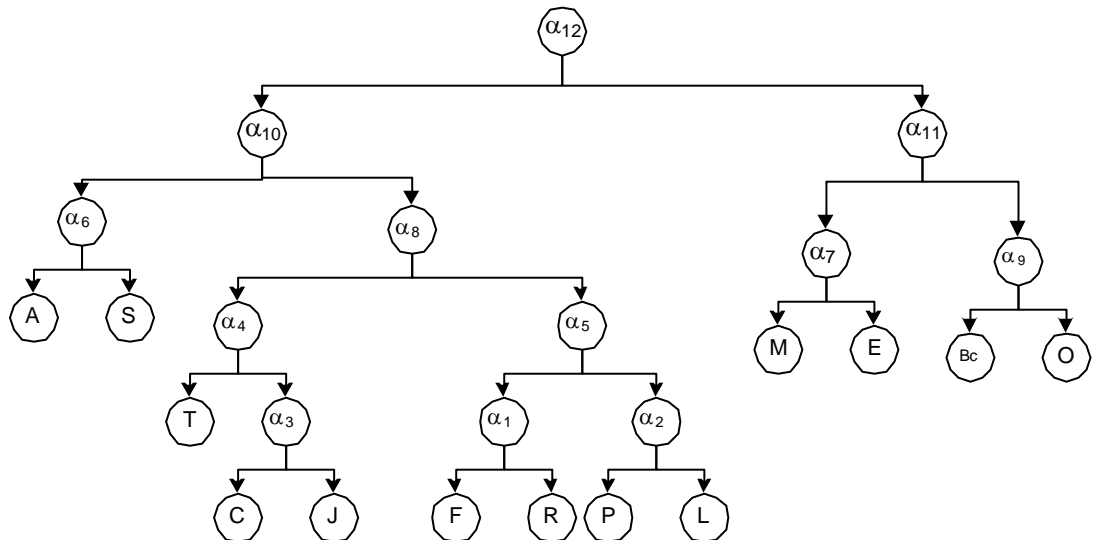
Status	Char	Freq	Code	Dir_tree
x	Blanco	5		dir (Bco)
x	O	5		dir (O)
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
x	$\alpha_4$	4		dir ( $\alpha_4$ )
x	$\alpha_5$	4		dir ( $\alpha_5$ )
x	$\alpha_6$	6		dir ( $\alpha_6$ )
x	$\alpha_7$	8		dir ( $\alpha_7$ )
x	$\alpha_8$	8		dir ( $\alpha_8$ )
x	$\alpha_9$	10		dir ( $\alpha_9$ )
	$\alpha_{10}$	14		dir ( $\alpha_{10}$ )
	$\alpha_{11}$	18		dir ( $\alpha_{11}$ )



Paso 12

Tomo los próximos dos caracteres de menor frecuencia,  $\alpha_{10}$  y  $\alpha_{11}$ . Armo un sub-árbol con raíz " $\alpha_{12}$ " en memoria con ambos nodos  $\alpha_{10}$  como hijo izq. y  $\alpha_{11}$  como hijo derecho. Actualizo la tabla de frecuencias con el nuevo nodo " $\alpha_{12}$ " y con su frecuencia como la suma de frecuencias de sus hijos, las direcciones del nodo raíz y el status de  $\alpha_{10}$  y  $\alpha_{11}$ .

Status	Char	Freq	Code	Dir_tree
x	Blanco	5		dir (Bco)
x	O	5		dir (O)
x	M	4		dir (M)
x	E	4		dir (E)
x	A	3		dir (A)
x	S	3		dir (S)
x	T	2		dir (T)
x	C	1		dir (C)
x	J	1		dir (J)
x	P	1		dir (P)
x	L	1		dir (L)
x	F	1		dir (F)
x	R	1		dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
x	$\alpha_4$	4		dir ( $\alpha_4$ )
x	$\alpha_5$	4		dir ( $\alpha_5$ )
x	$\alpha_6$	6		dir ( $\alpha_6$ )
x	$\alpha_7$	8		dir ( $\alpha_7$ )
x	$\alpha_8$	8		dir ( $\alpha_8$ )
x	$\alpha_9$	10		dir ( $\alpha_9$ )
x	$\alpha_{10}$	14		dir ( $\alpha_{10}$ )
x	$\alpha_{11}$	18		dir ( $\alpha_{11}$ )
	$\alpha_{12}$	32		dir ( $\alpha_{12}$ )



Paso 13

Al querer tomar los próximos dos caracteres de menor frecuencia, se observa que queda un único carácter con status null y que su frecuencia es la suma de todas las frecuencias de los caracteres originales. Por lo tanto se da por finalizado la generación del árbol.

**Compresión o Compactación de Datos.**

Este proceso comprende la lectura del archivo original, la búsqueda en la tabla de frecuencias de dicho carácter leído, y el acceso con la dirección del árbol a la hoja correspondiente a dicho carácter. A partir de ello se va ascendiendo por el árbol con el puntero al padre (utilizando para almacenar un 0 si asciendo desde un hijo izq. ó un 1 si asciendo del hijo derecho) hasta llegar al nodo raíz (padre en null). En ese momento se desapilará la pila en una estructura de salida, y se ingresará sólo para la primera lectura de cada carácter el código comprimido en la tabla de frecuencias, para optimizar el algoritmo.

Al finalizar este proceso la tabla de frecuencias quedará con la siguiente información:

Status	Char	Freq	Code	Dir_tree
x	Blanco	5	110	dir (Bco)
x	O	5	111	dir (O)
x	M	4	100	dir (M)
x	E	4	101	dir (E)
x	A	3	000	dir (A)
x	S	3	001	dir (S)
x	T	2	0100	dir (T)
x	C	1	01010	dir (C)
x	J	1	01011	dir (J)
x	P	1	01110	dir (P)
x	L	1	01111	dir (L)
x	F	1	01100	dir (F)
x	R	1	01101	dir (R)
x	$\alpha_1$	2		dir ( $\alpha_1$ )
x	$\alpha_2$	2		dir ( $\alpha_2$ )
x	$\alpha_3$	2		dir ( $\alpha_3$ )
x	$\alpha_4$	4		dir ( $\alpha_4$ )
x	$\alpha_5$	4		dir ( $\alpha_5$ )
x	$\alpha_6$	6		dir ( $\alpha_6$ )
x	$\alpha_7$	8		dir ( $\alpha_7$ )
x	$\alpha_8$	8		dir ( $\alpha_8$ )
x	$\alpha_9$	10		dir ( $\alpha_9$ )
x	$\alpha_{10}$	14		dir ( $\alpha_{10}$ )
x	$\alpha_{11}$	18		dir ( $\alpha_{11}$ )
	$\alpha_{12}$	32		dir ( $\alpha_{12}$ )

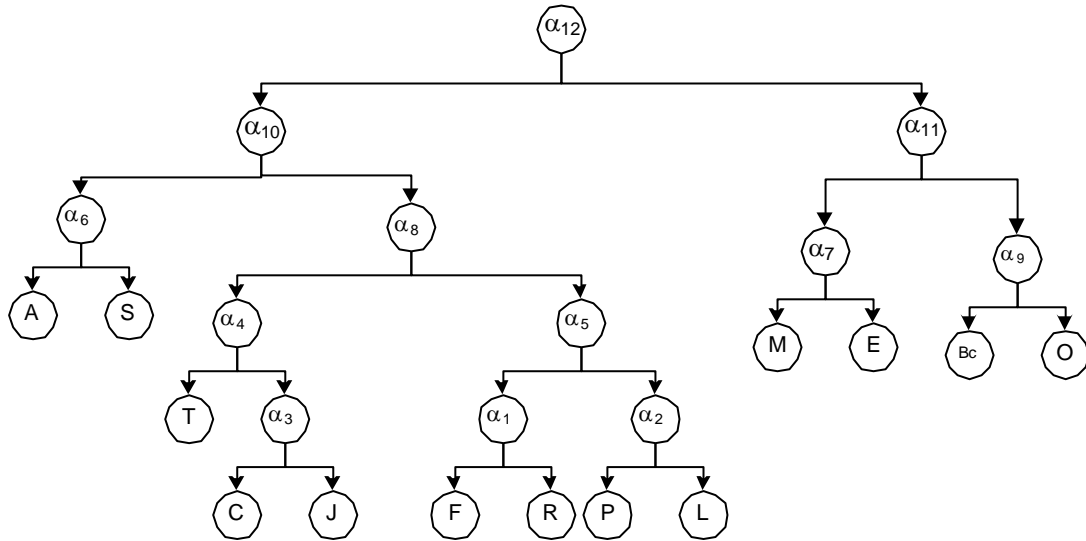
El archivo comprimido sería:

0100111100000100111001110010101111001111101010101110110001110011111111101010010  
1000001100110001101000001101

En total serían 107 bits, que en comparación con los 256 bits originales observamos una compactación de apróx. 42%.

### Descompresión o Descompactación de datos

Este proceso consiste en ir leyendo los bits del archivo compactado y accediendo desde la raíz del árbol ir descendiendo por el mismo, de la siguiente forma: si leo un bit 0 se deberá descender por el hijo izquierdo, si en cambio, leo un bit 1 se deberá descender por el hijo derecho. Este descenso será hasta llegar a una hoja (hijos en null), en dicho caso se informará su Id. Y se volverá a la raíz para seguir parseando los próximos bits.



0100 111 100 000 100 111 001 110 01010 111 100 111 110 101 01011 101 100 01110 01111 111  
 T O M A M O S \_ C O M O \_ E J E M P L O  
 110 101 001 0100 000 110 01100 01101 000 001 101  
 \_ E S T A \_ F R A S E