

# Estructuras de Datos no Lineales

## 1.5.3. Árboles B

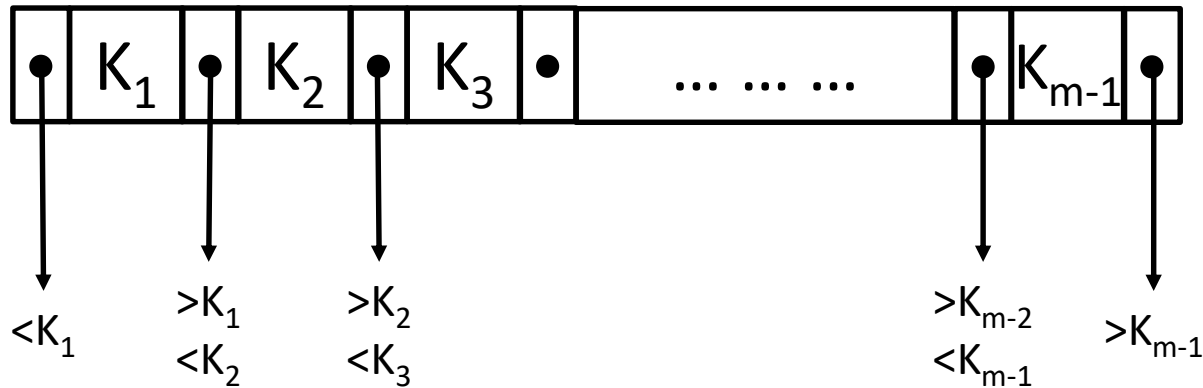
José Fidel Argudo Argudo  
José Antonio Alonso de la Huerta  
M<sup>a</sup> Teresa García Horcajadas



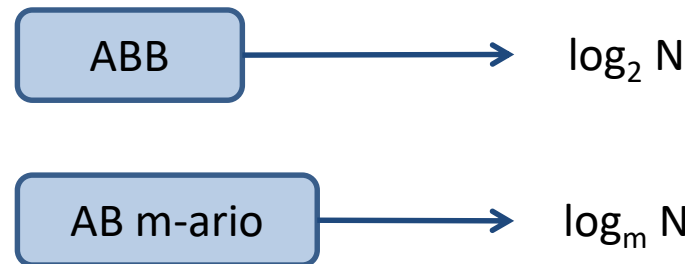
Versión 2.0

# Árboles B

- En enormes colecciones de datos (con millones de registros) almacenadas en memoria secundaria los tiempos de búsqueda con ABB se disparan.
  - En cada paso del algoritmo hay que acceder a un dato almacenado en la memoria externa.
  - El tiempo de acceso es del orden de millones de veces superior que en memoria principal.
- **Objetivo: Reducir los accesos a memoria secundaria.**
  - Generalización del concepto de ABB: Se almacenan varios elementos en un mismo nodo.
  - Lo ideal es que un nodo ocupe un bloque de memoria secundaria.



Localización de un registro en un fichero (accesos a bloques)



El valor  $m$  no puede ser arbitrariamente grande porque:

- Un nodo del árbol será mayor que un bloque y será necesario más de un acceso a memoria secundaria para leerlo.
- Es más lento leer y procesar un nodo más grande.

# Árbol B

1. Un nodo que no es hoja con  $k$  claves tiene  $k+1$  hijos.
2. Cada nodo tiene como máximo  $m$  hijos y  $m$  es el orden del árbol.
3. Cada nodo, excepto la raíz, contiene por lo menos  $\lfloor (m-1)/2 \rfloor$  claves y, por supuesto, no más de  $m-1$  claves.
4. La raíz tiene como mínimo una clave y dos hijos (al menos que sea hoja).
5. Todas las hojas aparecen en el mismo nivel.

**EL ORDEN ES SIEMPRE LA PARTE ENTERA**

# Inserción en un árbol B

Insertar la siguiente secuencia de valores en un árbol B de orden 4.

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

Como tiene orden 4, el numero de claves por nodos es 3 ( $m-1$ )

# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

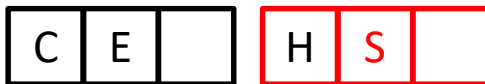
C	E	H
---	---	---



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

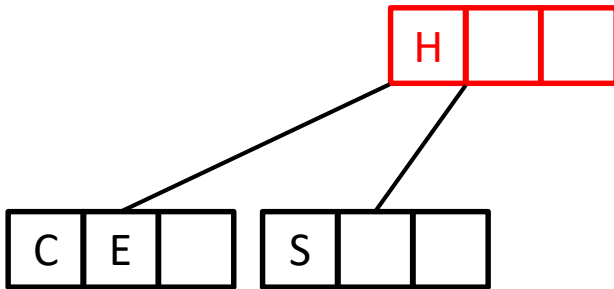
División



# Inserción en un árbol B

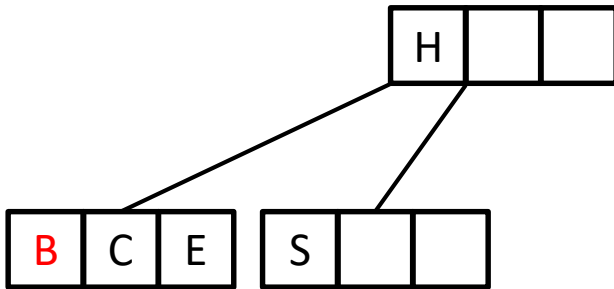
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

Promoción



# Inserción en un árbol B

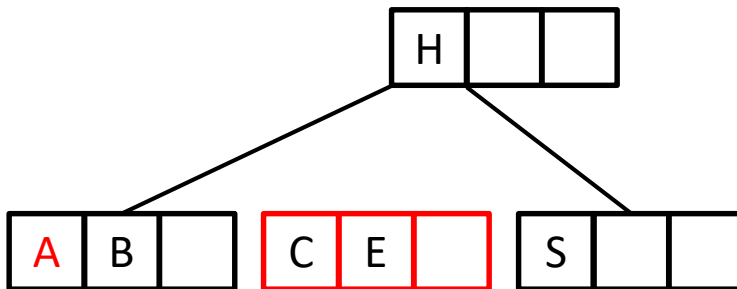
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

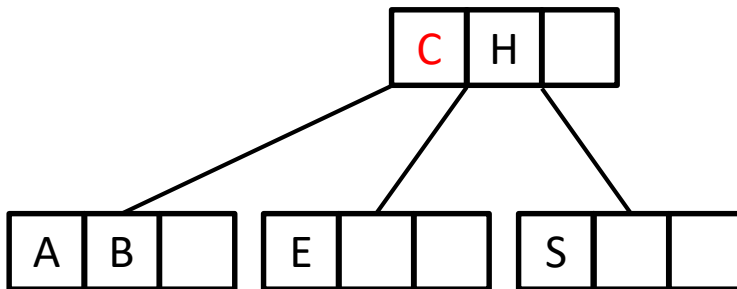
División



# Insertión en un árbol B

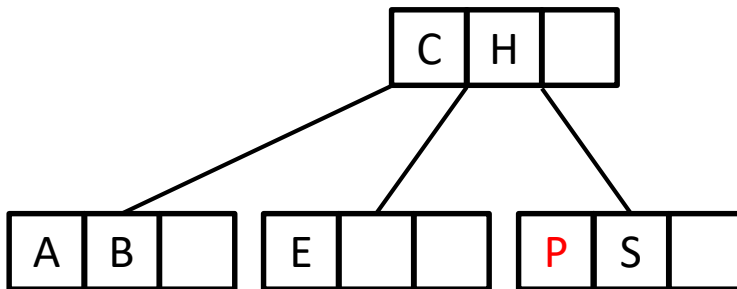
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

Promoción



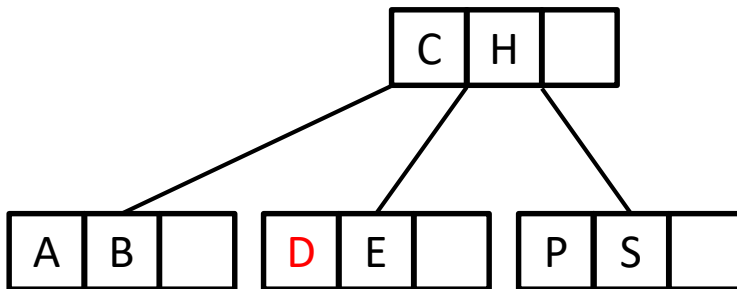
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



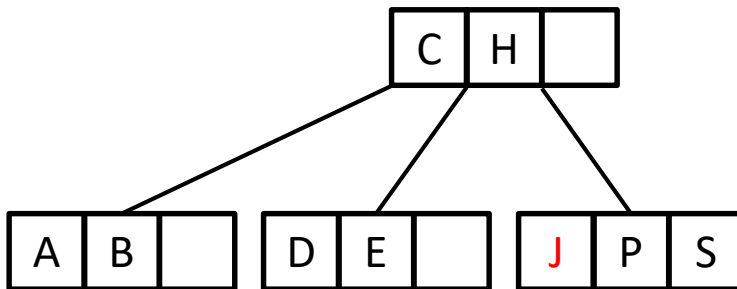
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

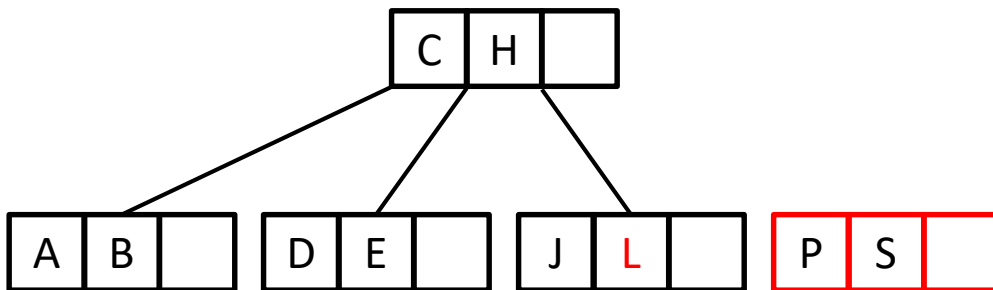




# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

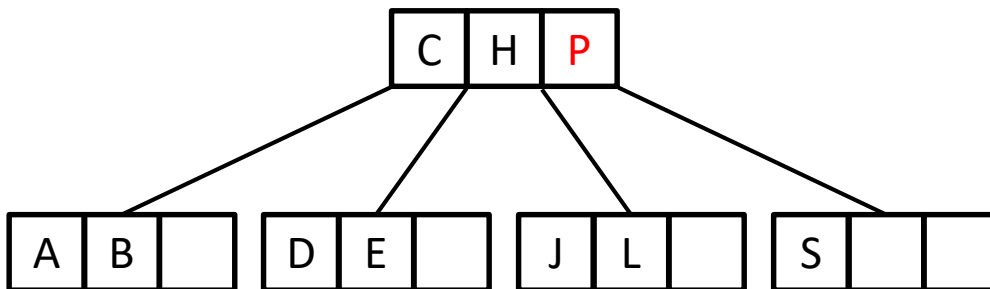
División



# Inserción en un árbol B

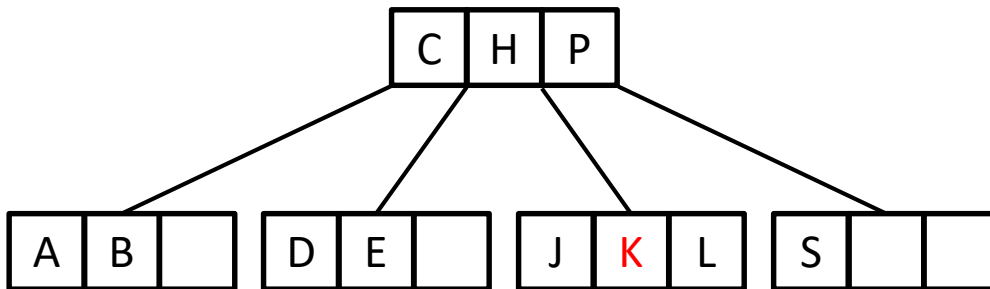
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

Promoción



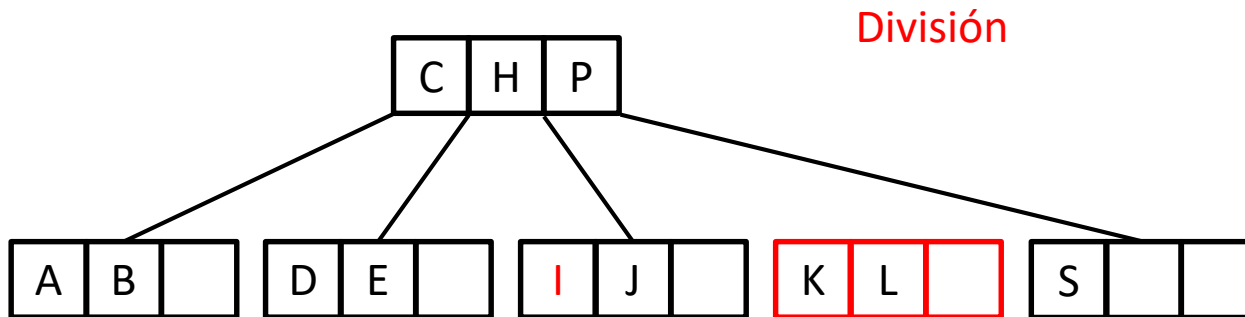
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

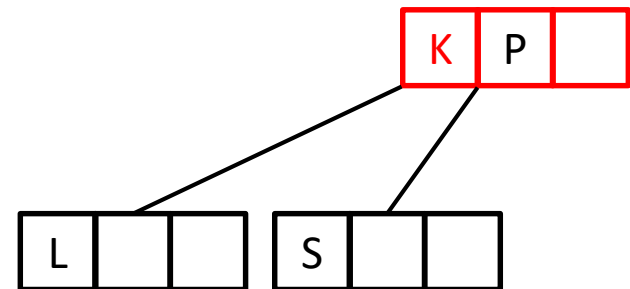
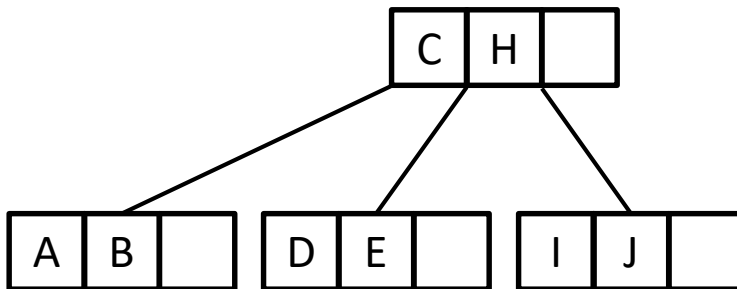
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

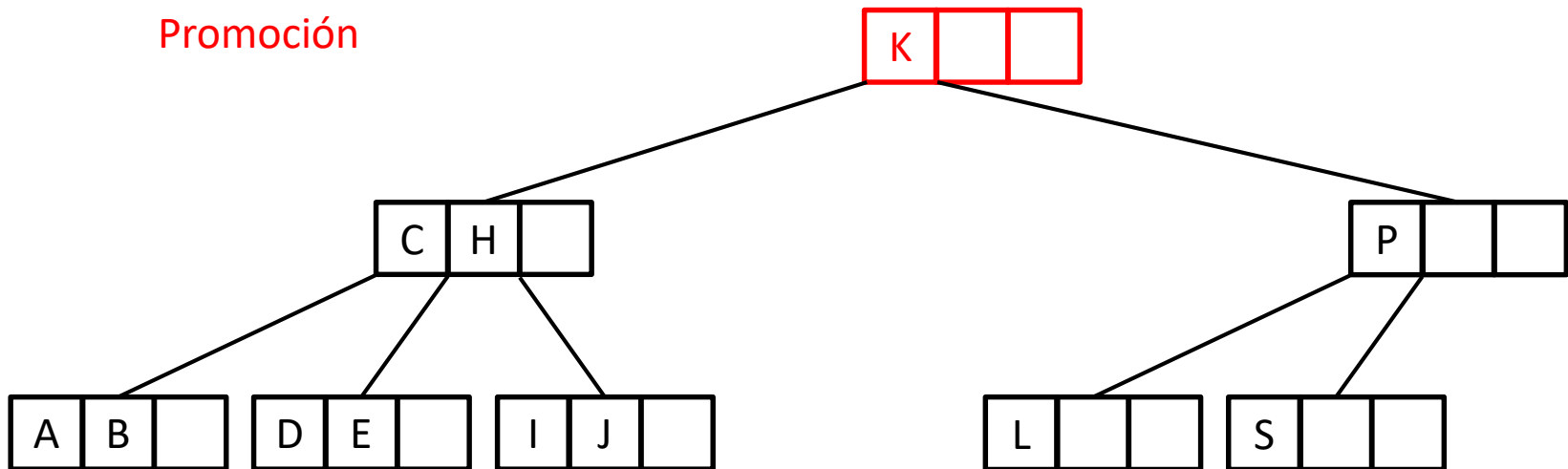
Promoción y división



# Inserción en un árbol B

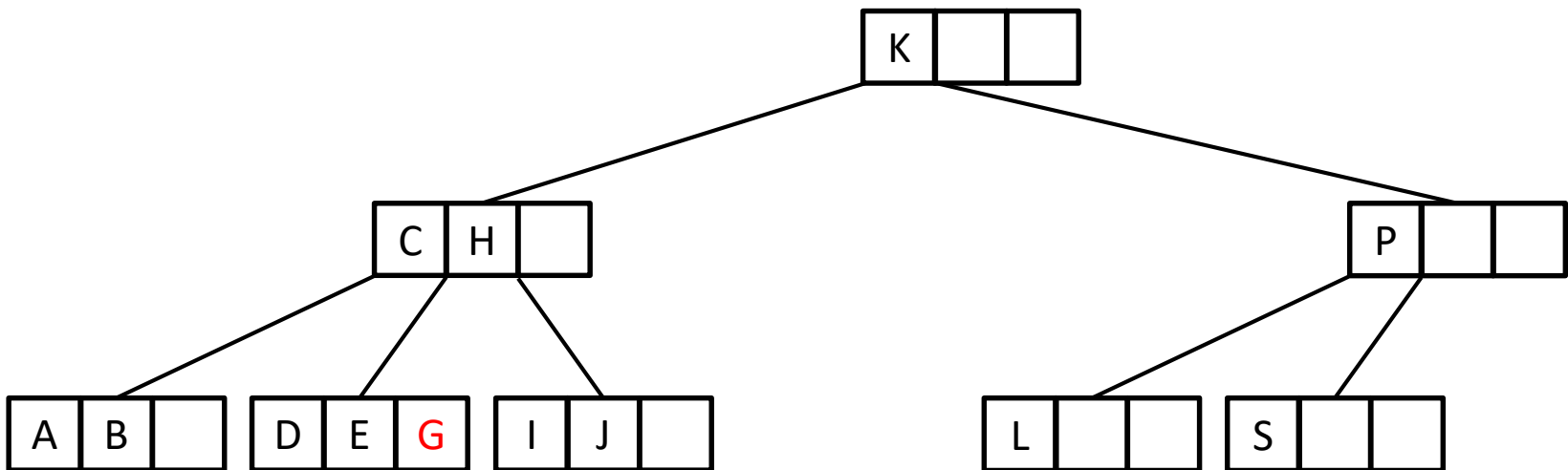
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y

Promoción



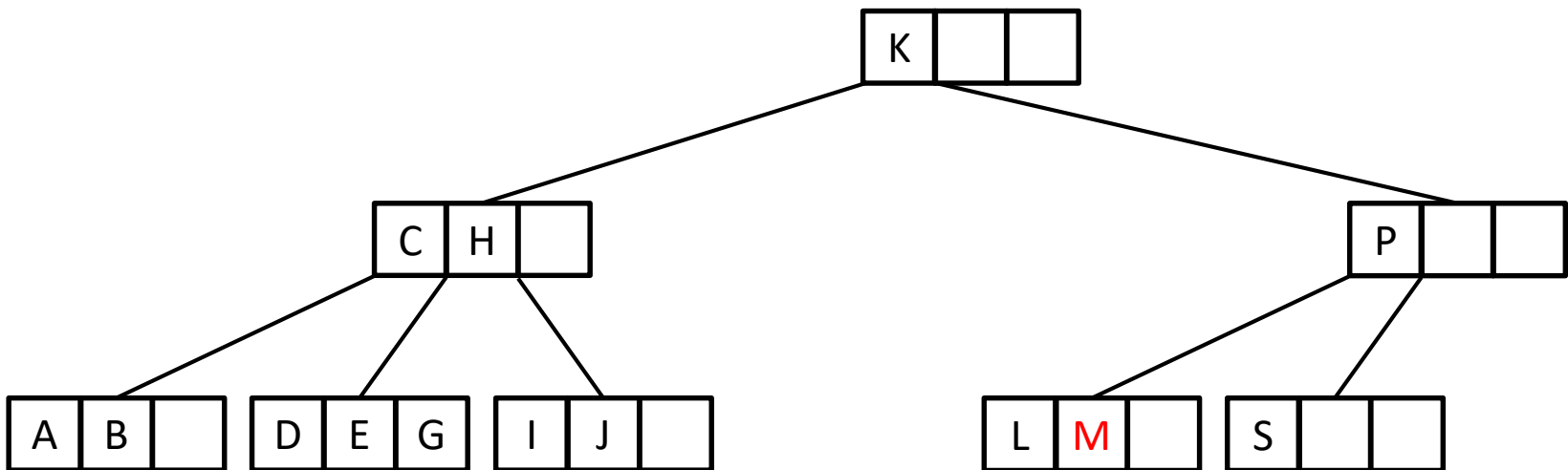
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Inserción en un árbol B

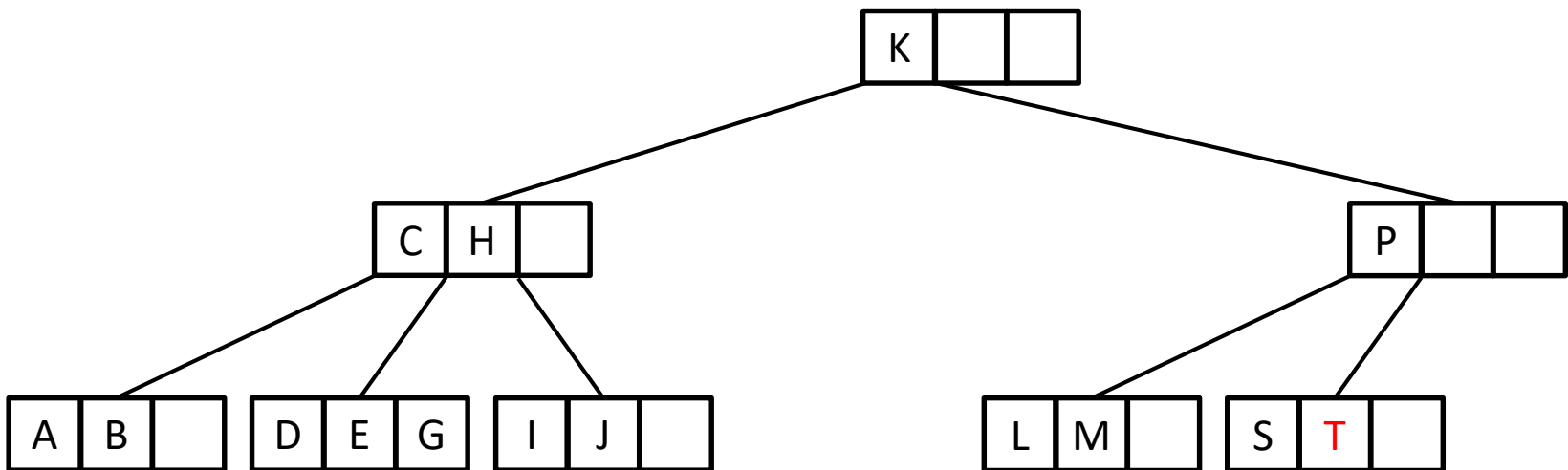
C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y





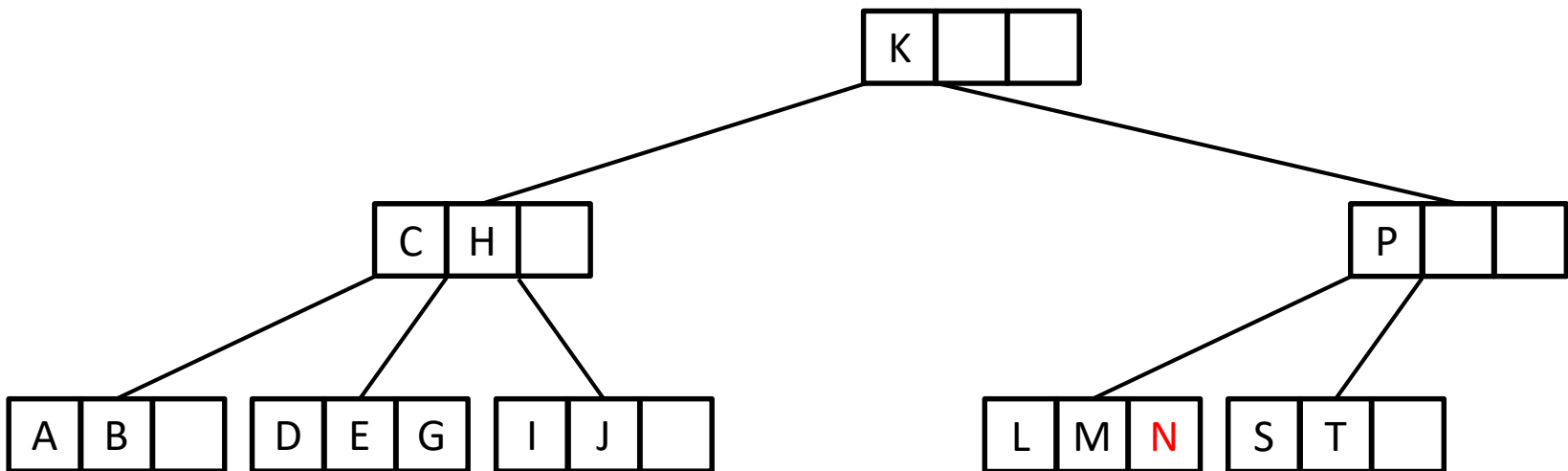
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



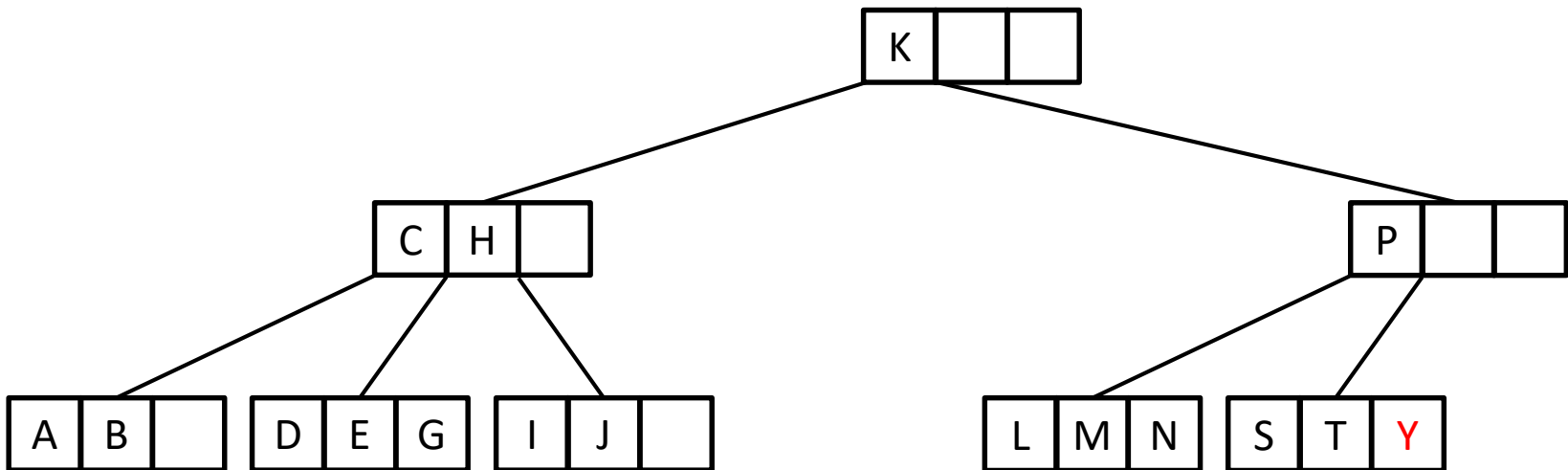
# Inserción en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



# Insertión en un árbol B

C, E, H, S, B, A, P, D, J, L, K, I, G, M, T, N, Y



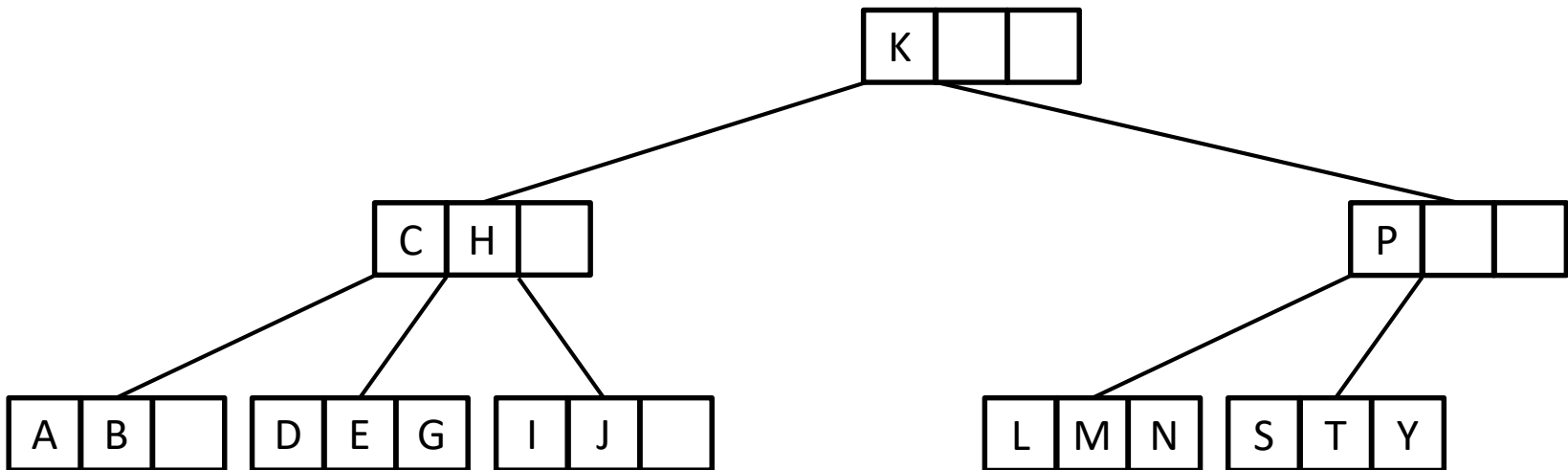
# Eliminación en un árbol B

1. Si la clave que se va a eliminar no está en una hoja, intercambiarla con su sucesor más cercano según el orden lineal de las claves, el cuál se encontrará siempre en una hoja.
2. Eliminar la clave.
3. Si el nodo del que se ha suprimido la clave contiene por lo menos el número mínimo de claves, se termina.
4. Si este nodo contiene una menos de las claves mínimas, comprobar los hermanos izquierdo y derecho.
  - a. Si existe un hermano que tenga más del número mínimo de claves, redistribuir las claves con este hermano.
  - b. Si no existe un hermano que tenga más del número mínimo de claves, se concatenan los dos nodos junto con la clave del nodo padre que separa ambos.
5. Si se han concatenado dos nodos, volver a aplicar al nodo padre desde el paso 3, excepto cuando el padre sea el nodo raíz, en cuyo caso, el árbol queda con un nivel menos.

# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

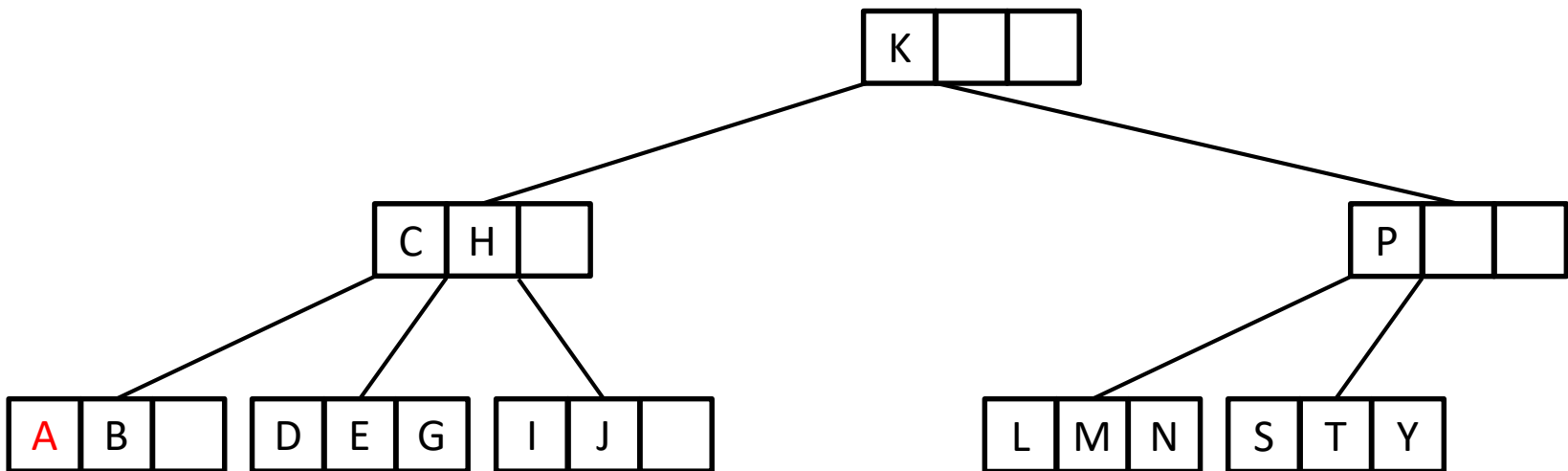
A, H, B, J, G, I, E



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

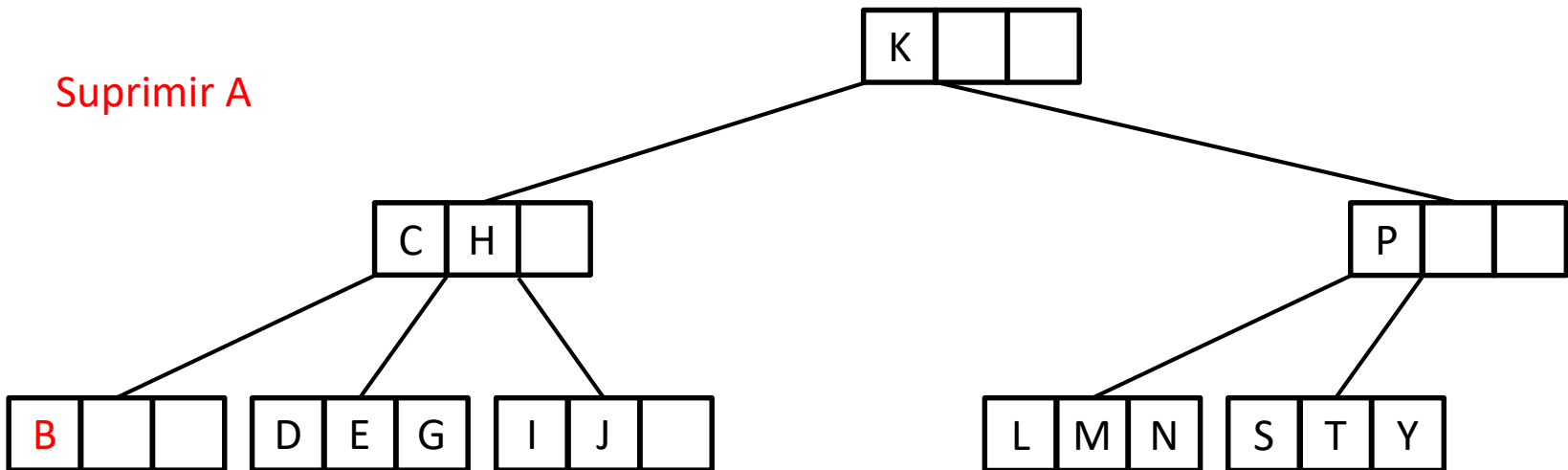


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

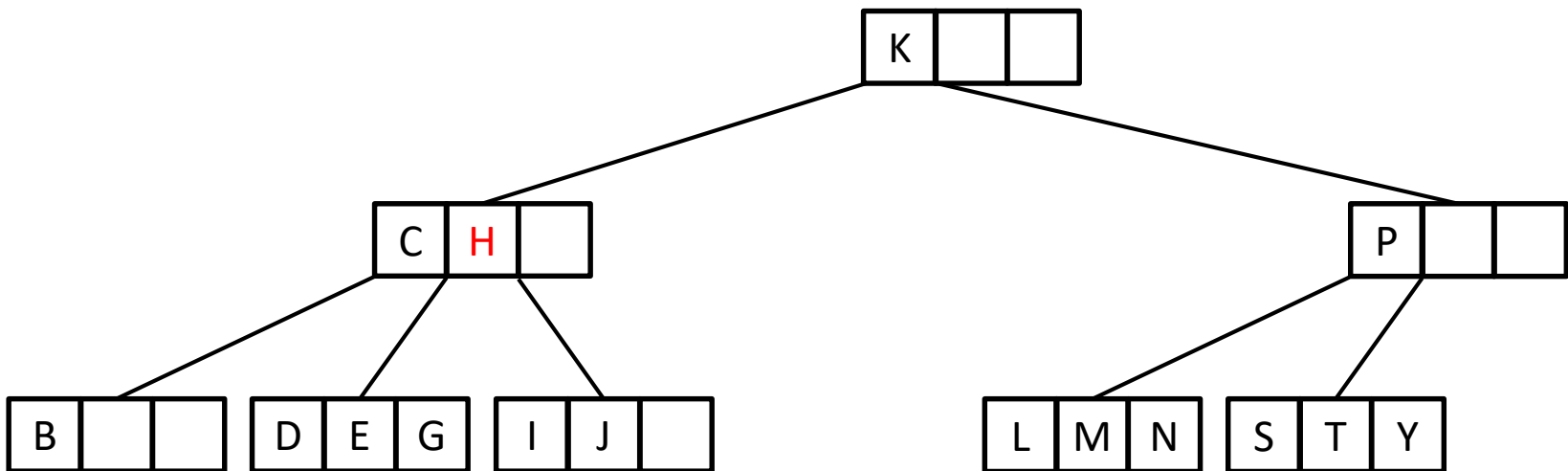
Suprimir A



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E



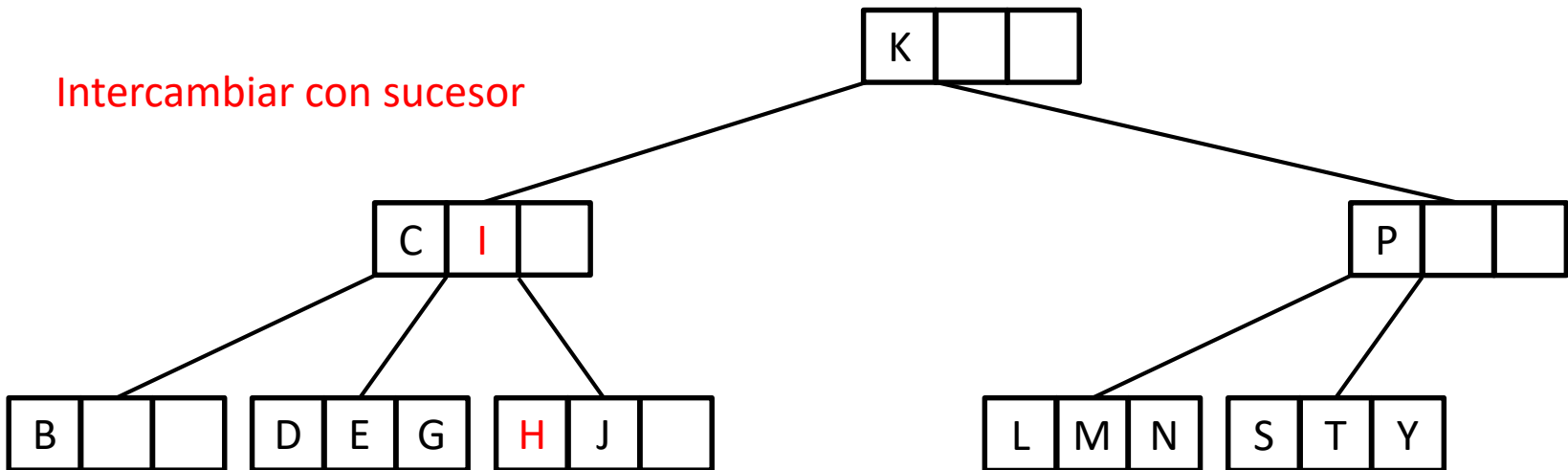


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

Intercambiar con sucesor

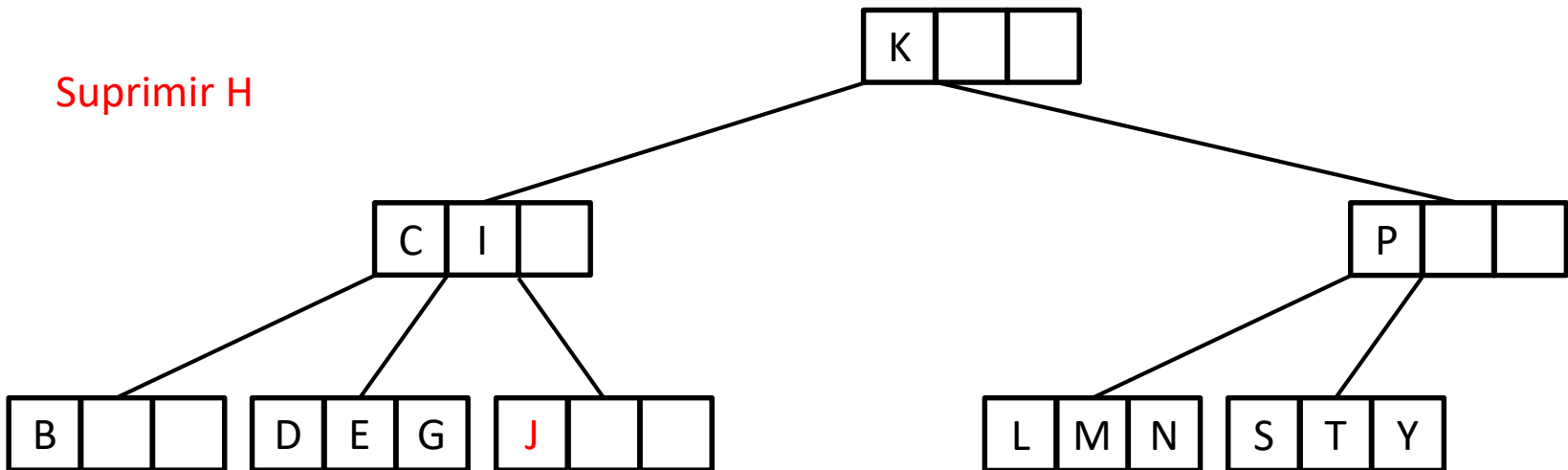


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

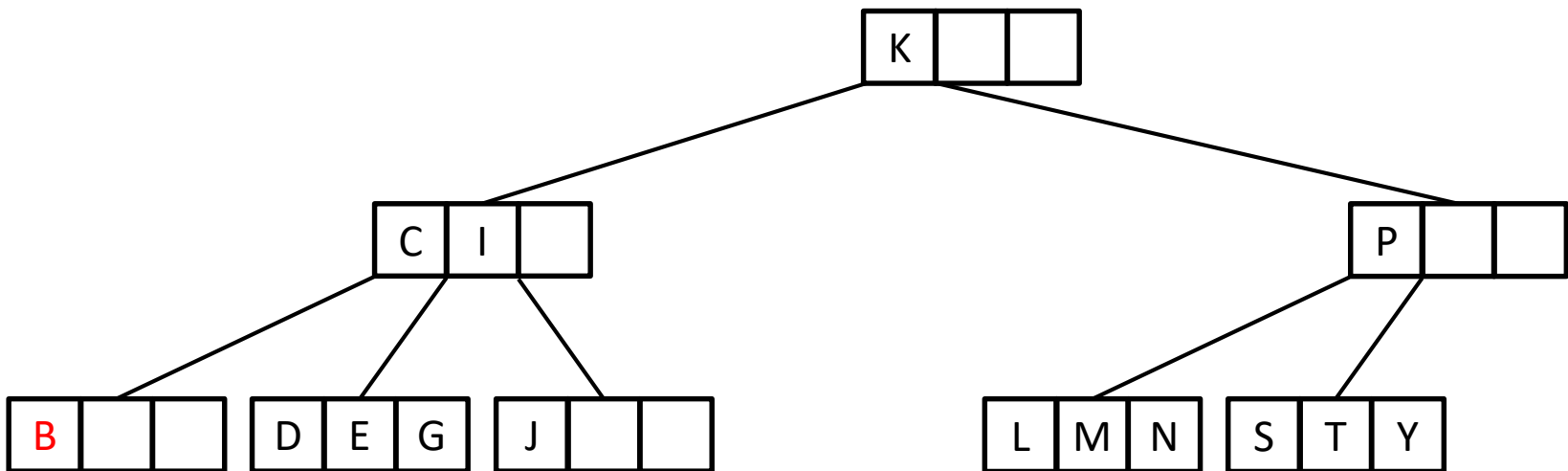
Suprimir H



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

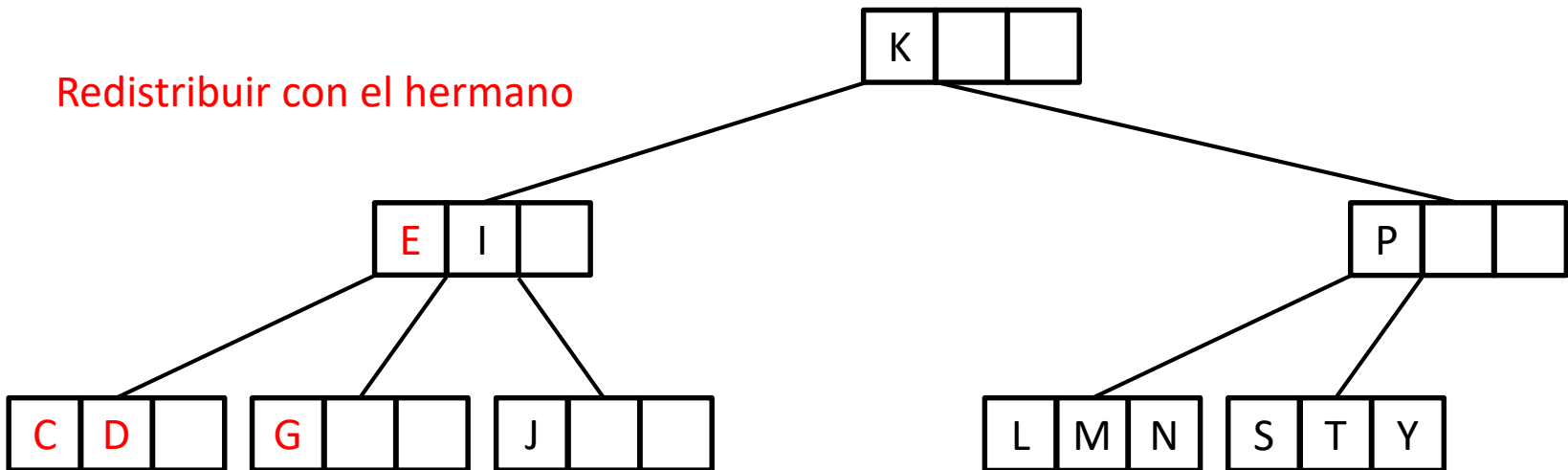


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

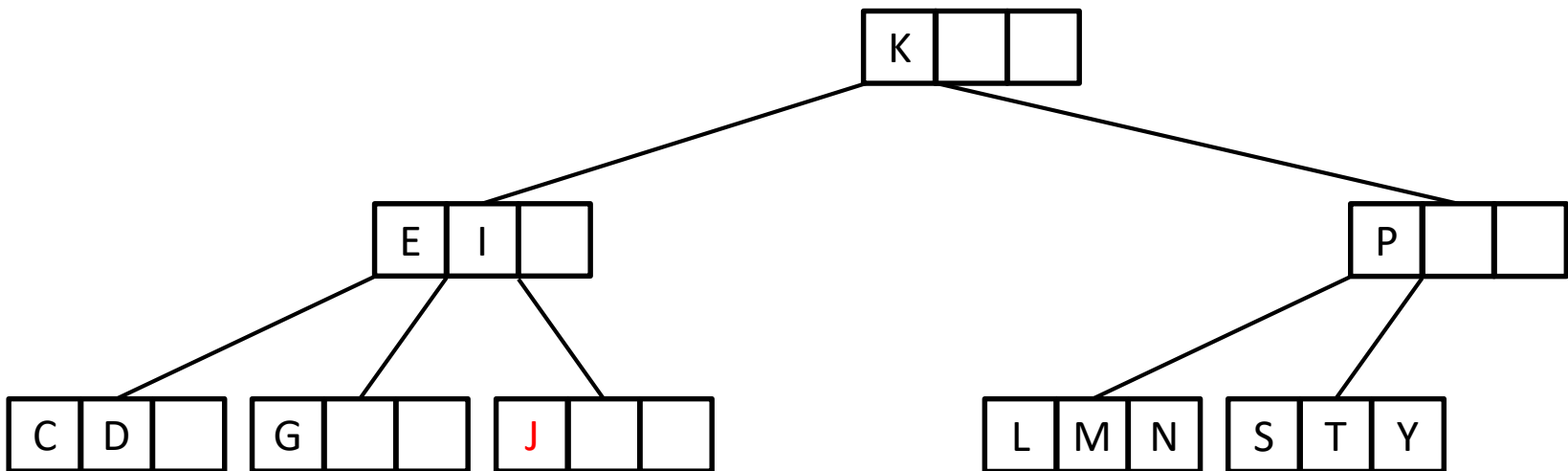
Redistribuir con el hermano



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

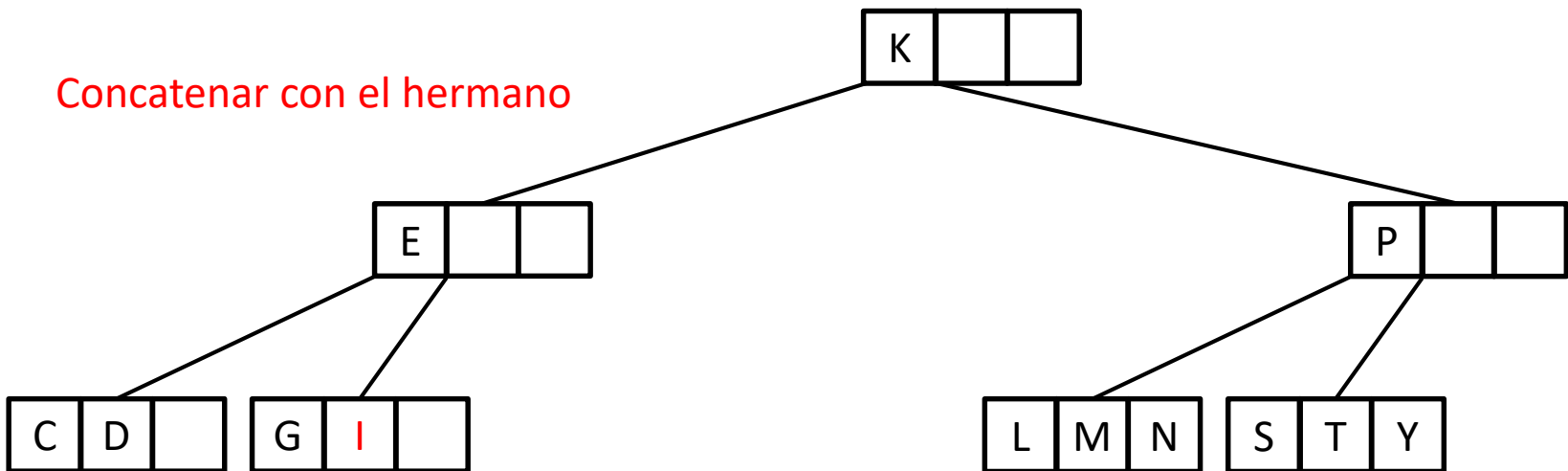


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

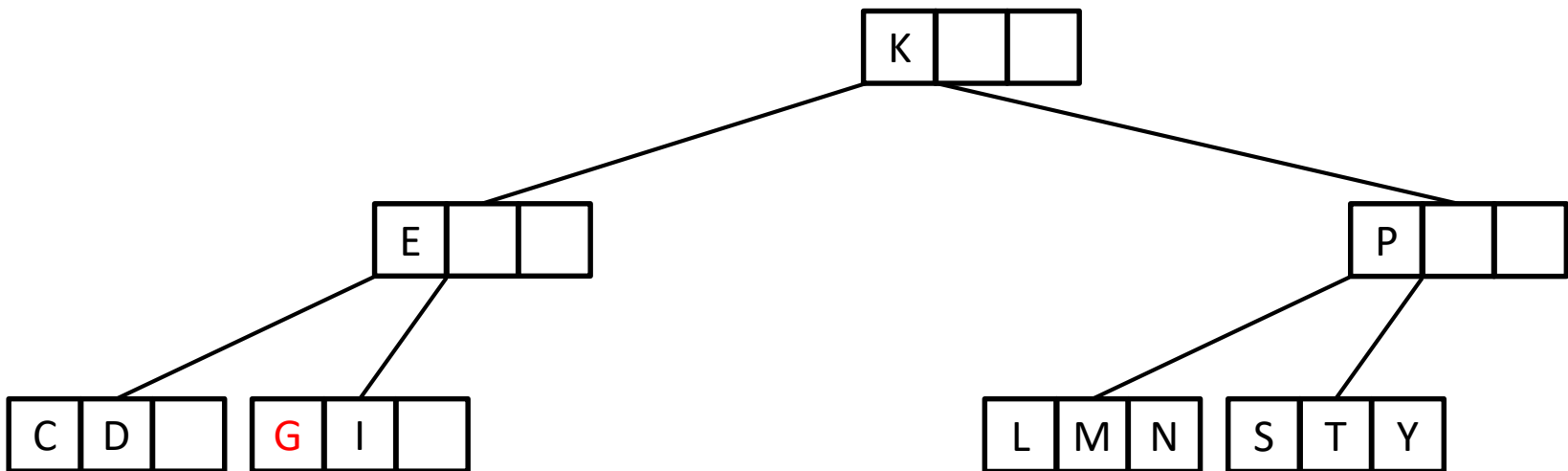
Concatenar con el hermano



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

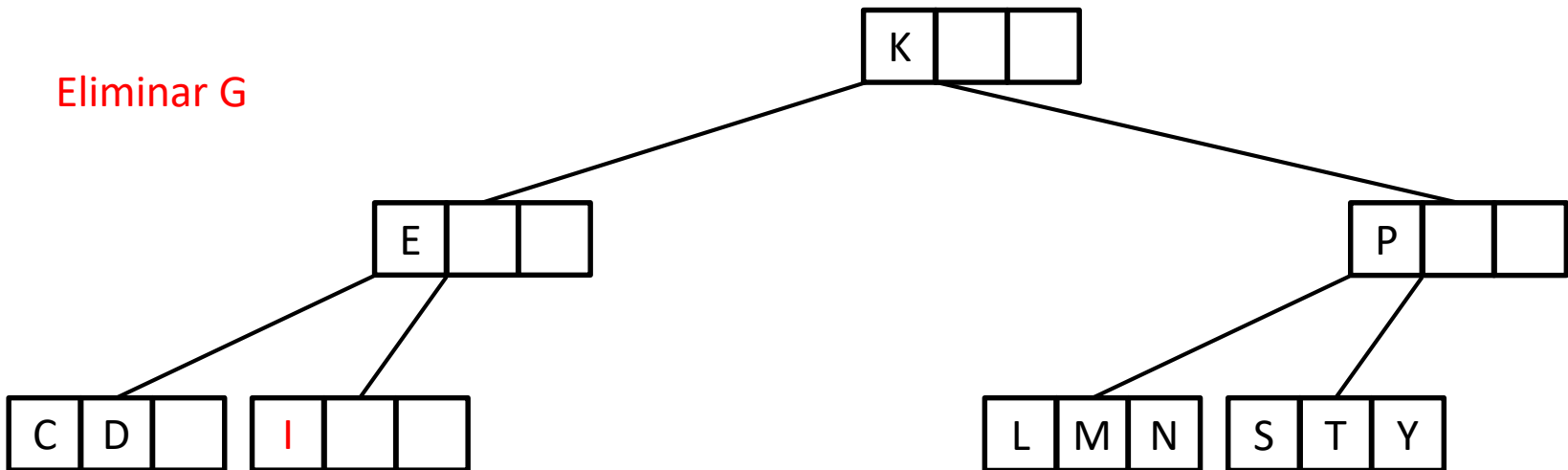


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

Eliminar G

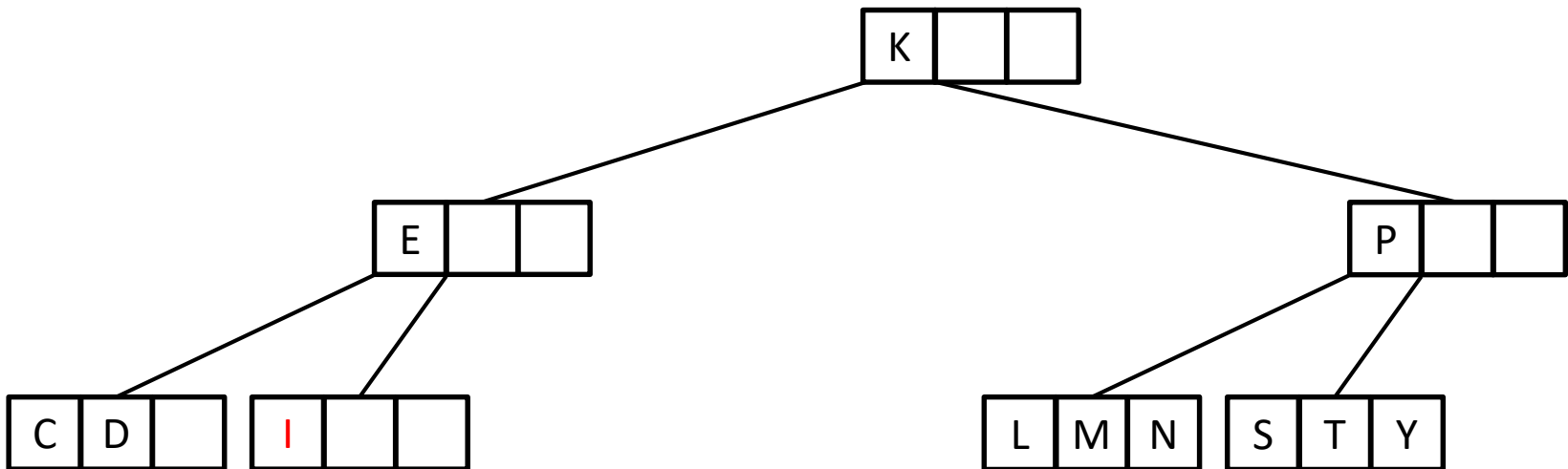




# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

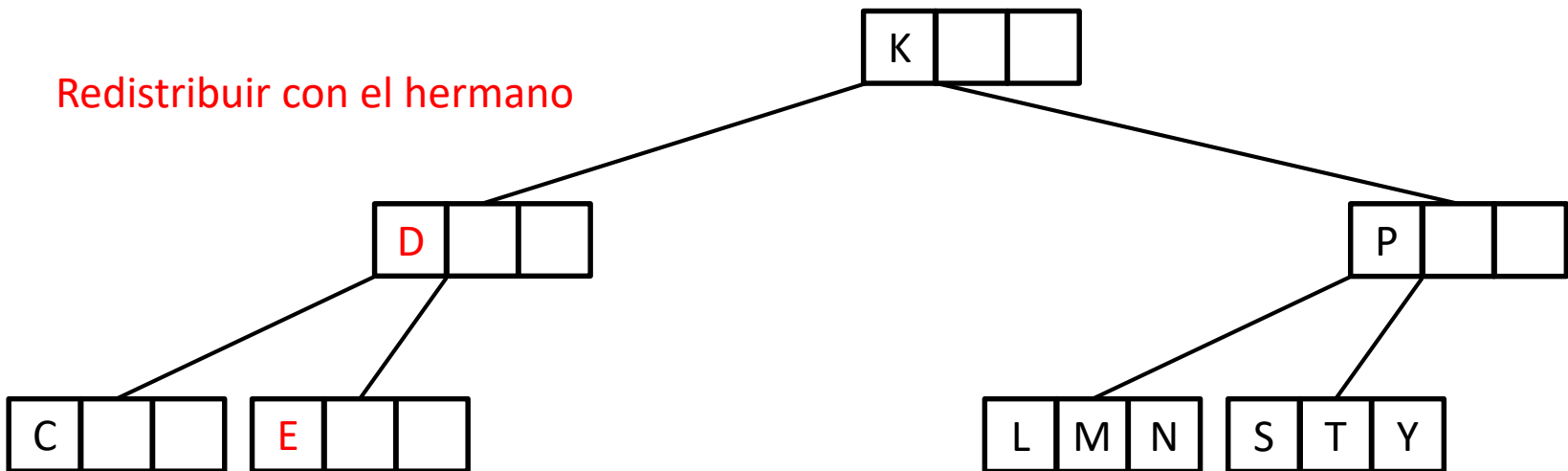


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

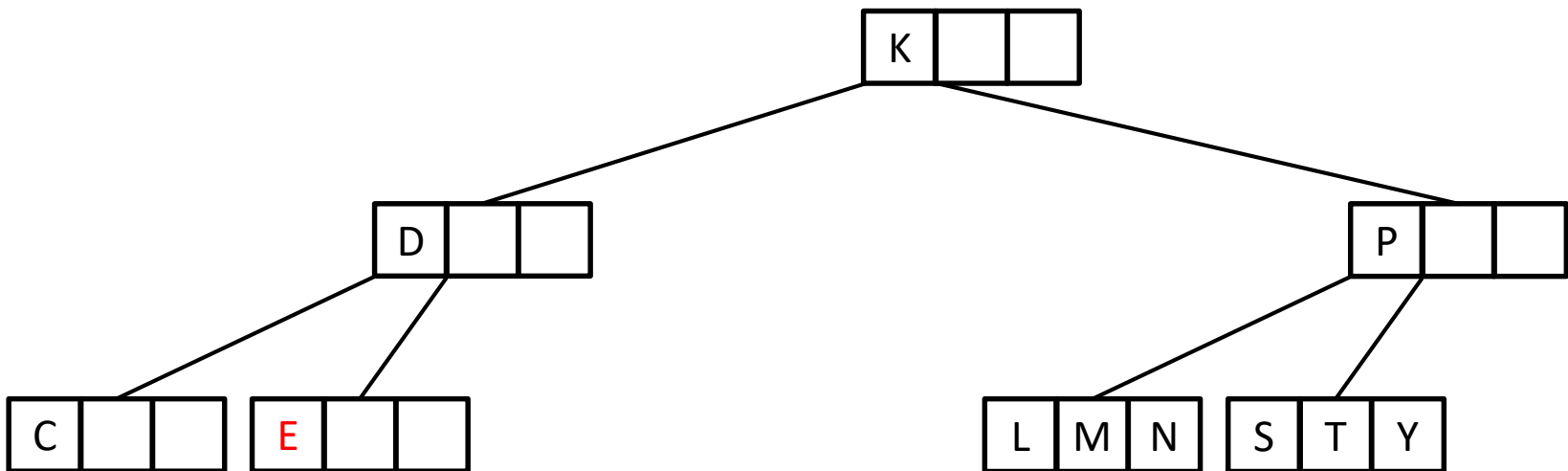
Redistribuir con el hermano



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

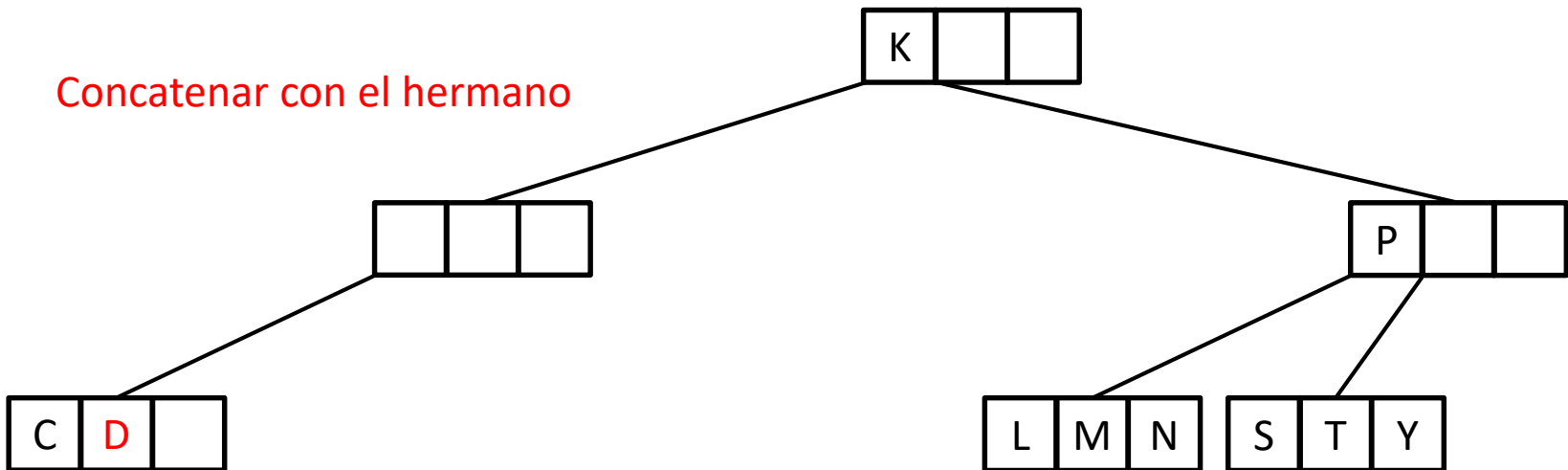


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

Concatenar con el hermano

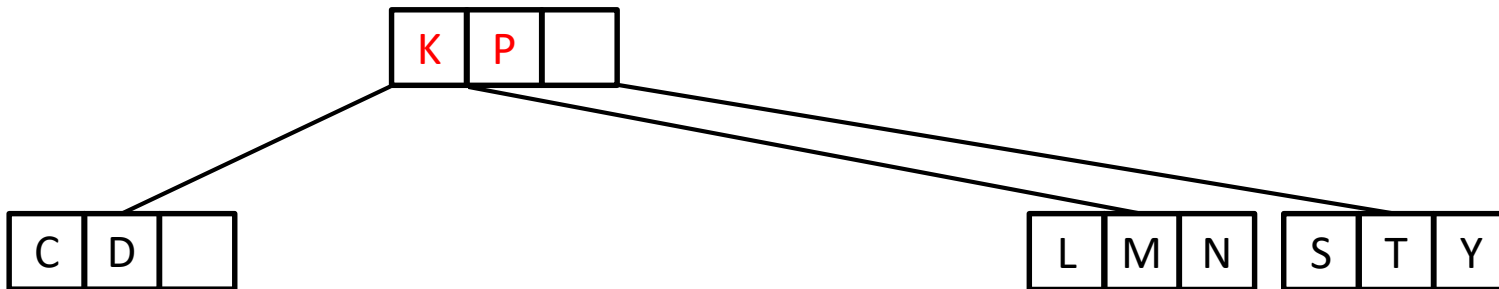


# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

Concatenar con el hermano



# Eliminación en un árbol B

Eliminar la siguiente secuencia de valores:

A, H, B, J, G, I, E

