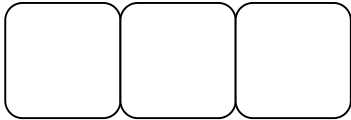


# **Exemplo de Manipulação de Árvore B+**

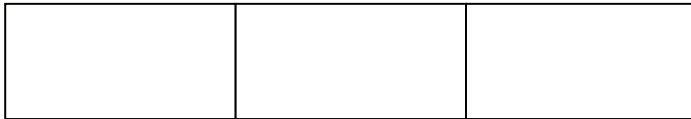
Prof. Joaquim Quinteiro Uchôa

DAC/UFLA - 2022



**Nós não-folha  
(índices)**

***Para este exemplo,  
adotaremos que cada nó  
suporta no mínimo 1 e no  
máximo 2 chaves***



**Nós folha (dados)  
(Sequence Set)**

**Sequência de Inserção:**

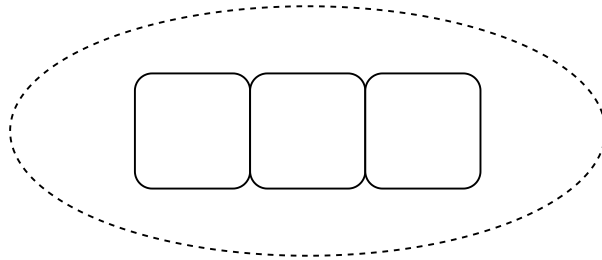
DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

**Remover após Inserção:**

DY, FO, FC, EC, EH

### Sequência de Inserção:

**DY**, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY



Árvore de índices inicialmente vazia,  
folha é também raiz.  
Nesse caso, numa implementação, isso  
pode ser verificado de diversas formas  
(índice nulo, folha ser raiz, etc.).



**Sequência de Inserção:**

*DY, ER*, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

DY	ER	
----	----	--

**Sequência de Inserção:**

*DY, ER, FO*, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

DY	ER	FO
----	----	----

### Sequência de Inserção:

**DY, ER, FO**, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

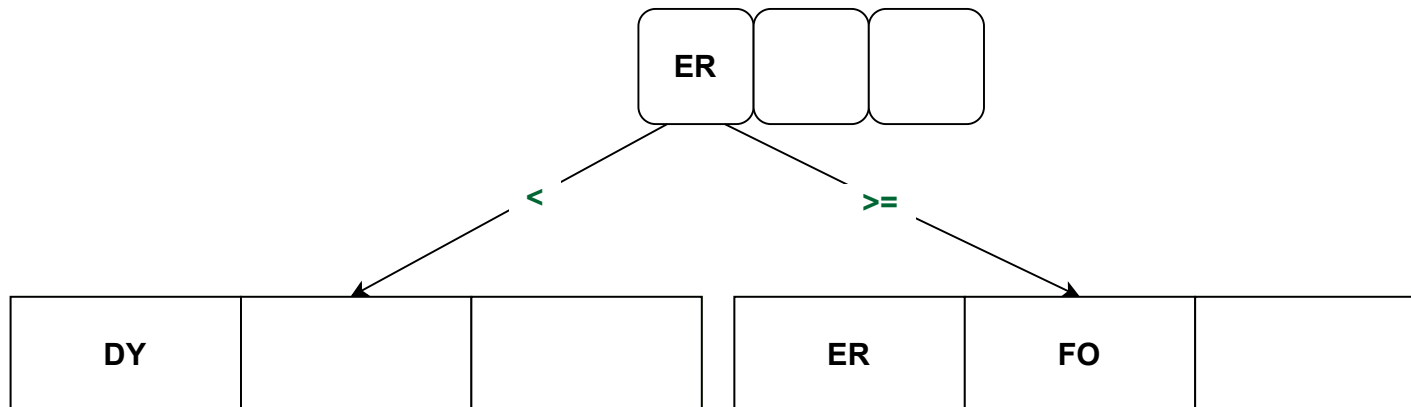
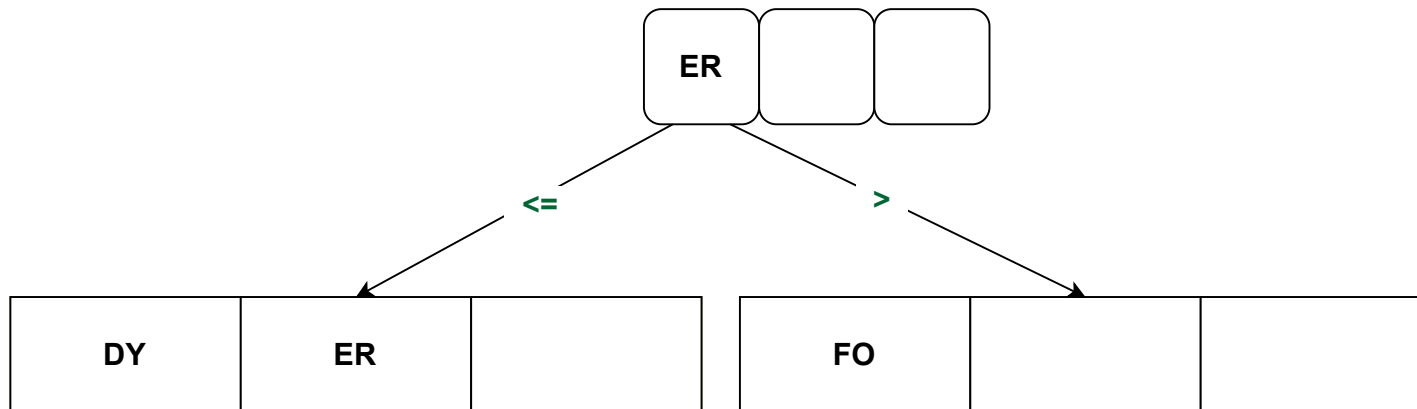
Para inserir o DE, o bloco está cheio, precisamos dividi-lo e, nesse momento, criar a árvore de índices

DY	ER	FO
----	----	----

### Sequência de Inserção:

**DY, ER, FO**, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

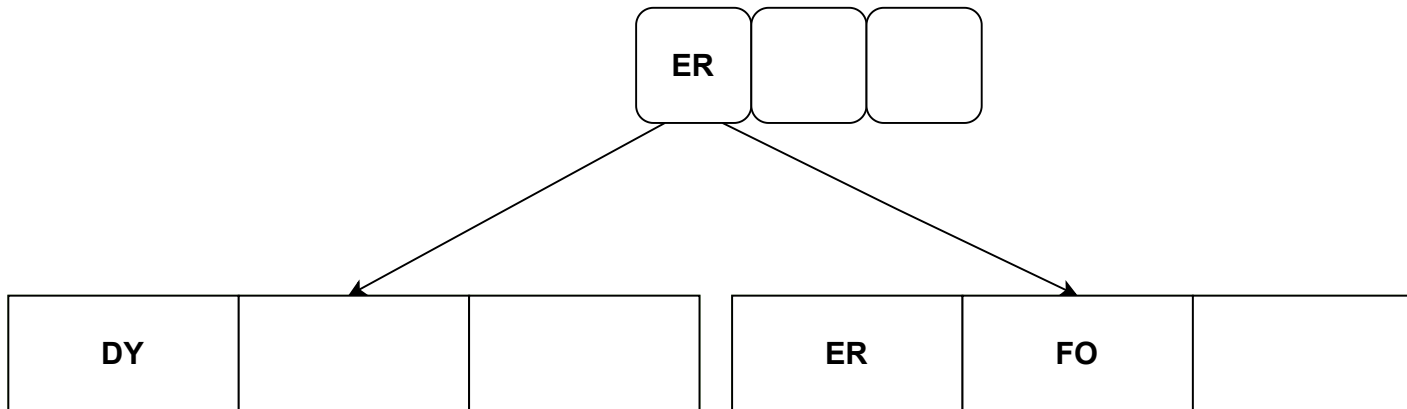
O valor que sobe é apenas a chave... Assim, o dado  
e a chave precisam continuar em um dos blocos...  
Temos duas possibilidades, pelo menos



### Sequência de Inserção:

**DY, ER, FO**, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY

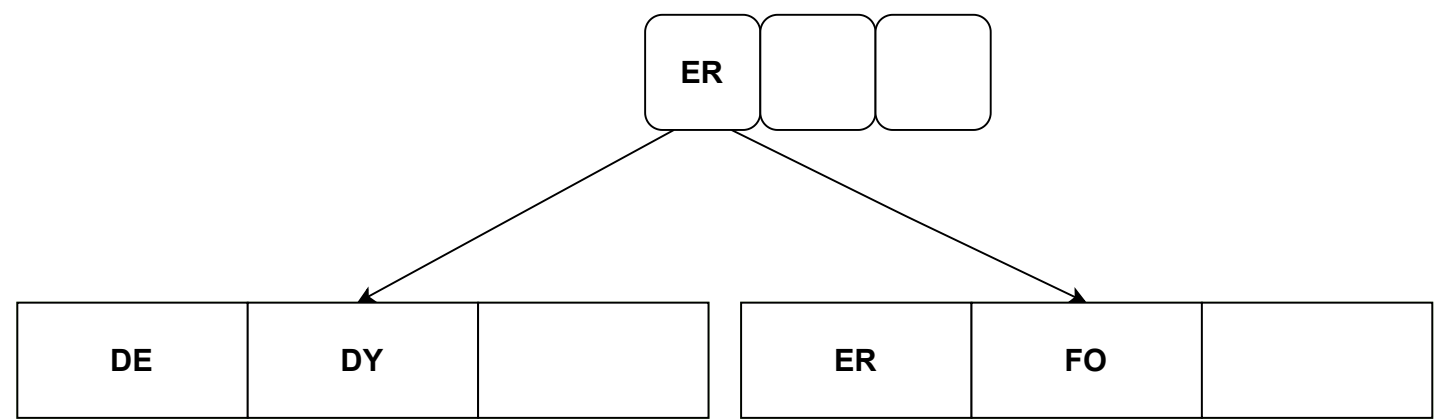
*A escolha pode depender do problema, mas em geral é arbitrária. E precisa ser adotada em todo o sistema.*





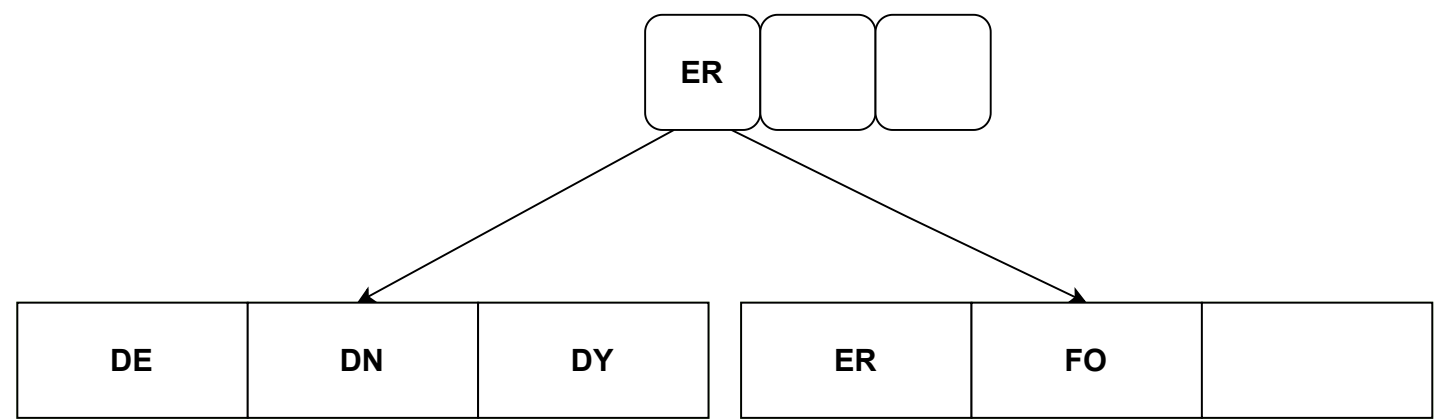
Sequência de Inserção:

DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY



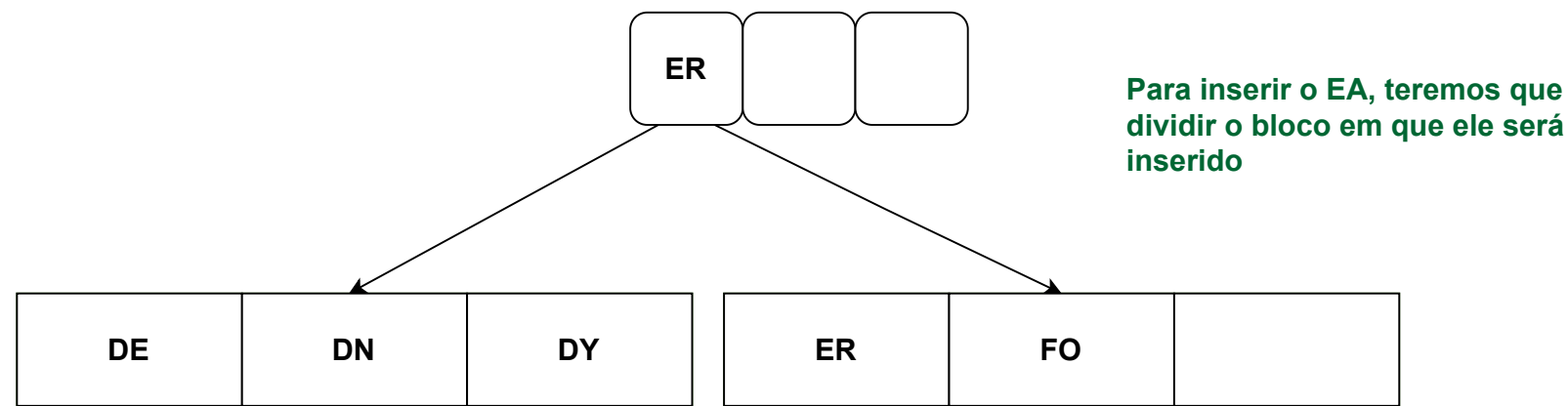
Sequência de Inserção:

DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY



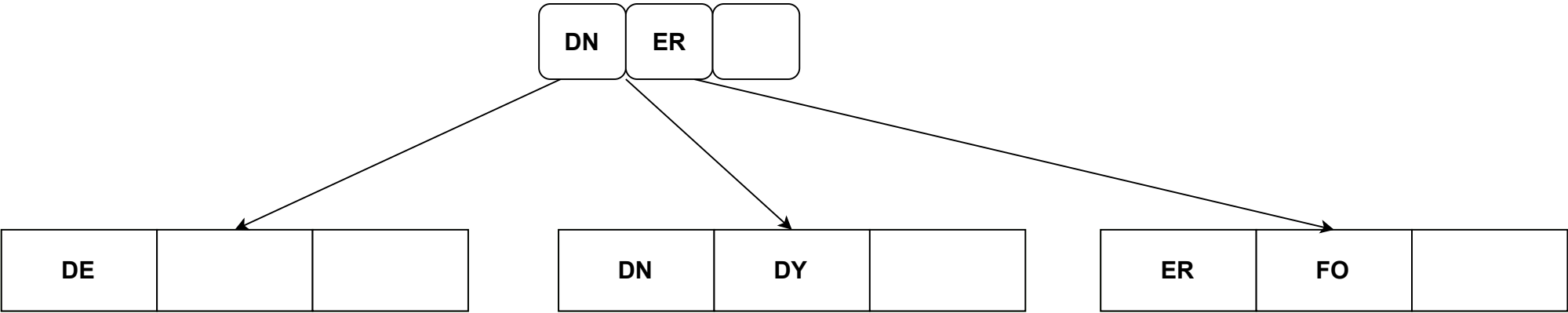
Sequência de Inserção:

DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY



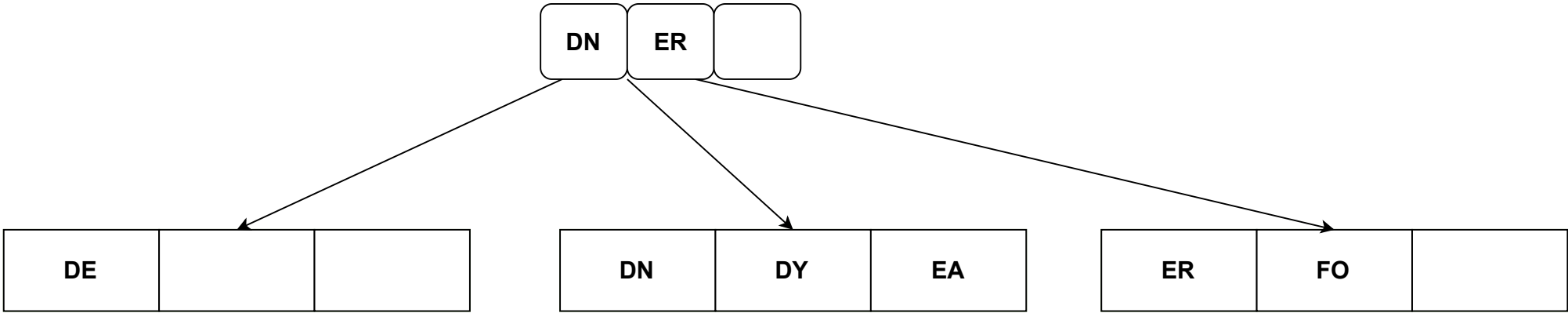
**Sequência de Inserção:**

*DY, ER, FO, DE, DN*, EA, EC, EH, FB, FC, GK, GZ, GY



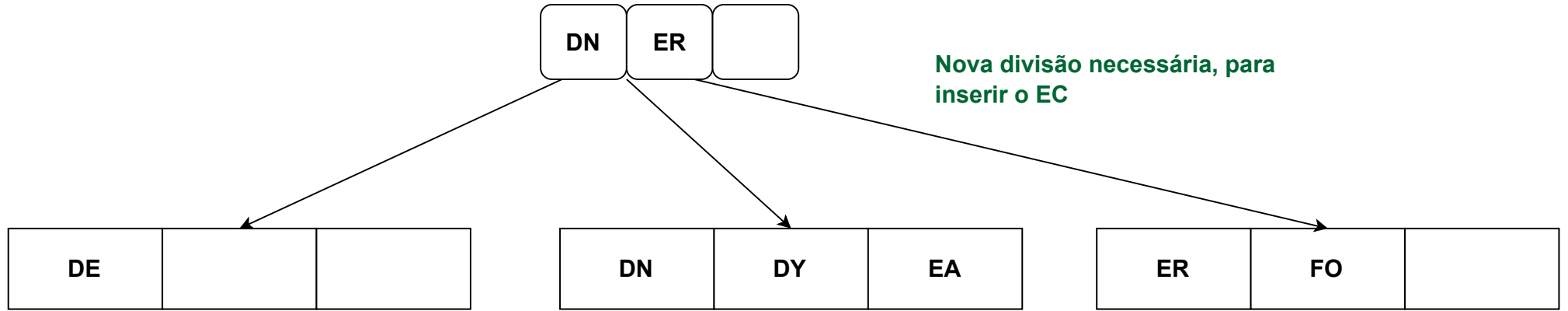
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA*, EC, EH, FB, FC, GK, GZ, GY



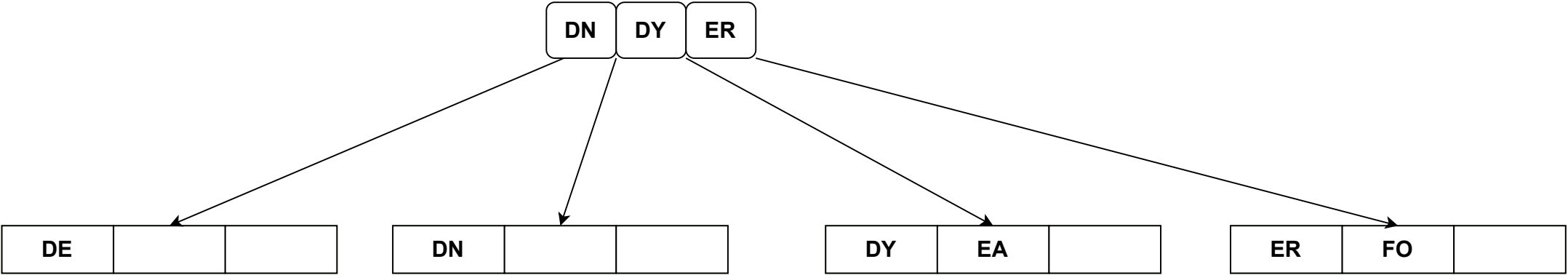
**Sequência de Inserção:**

**DY, ER, FO, DE, DN, EA**, EC, EH, FB, FC, GK, GZ, GY



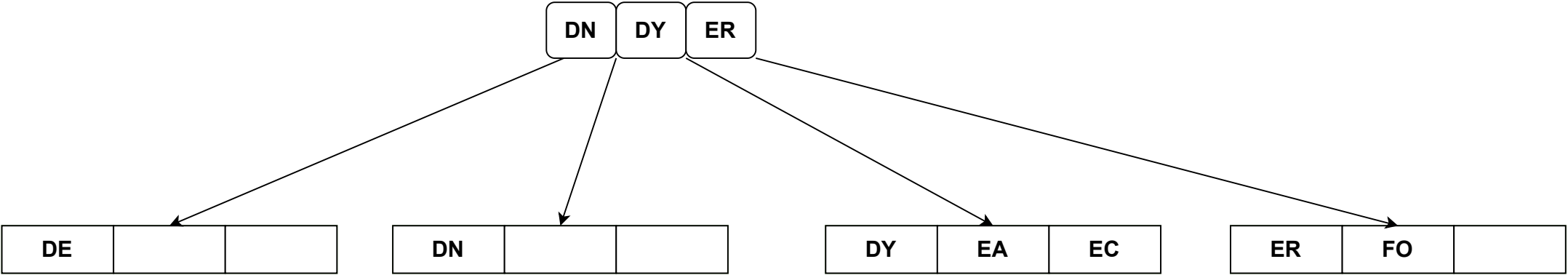
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA*, EC, EH, FB, FC, GK, GZ, GY



**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC*, EH, FB, FC, GK, GZ, GY

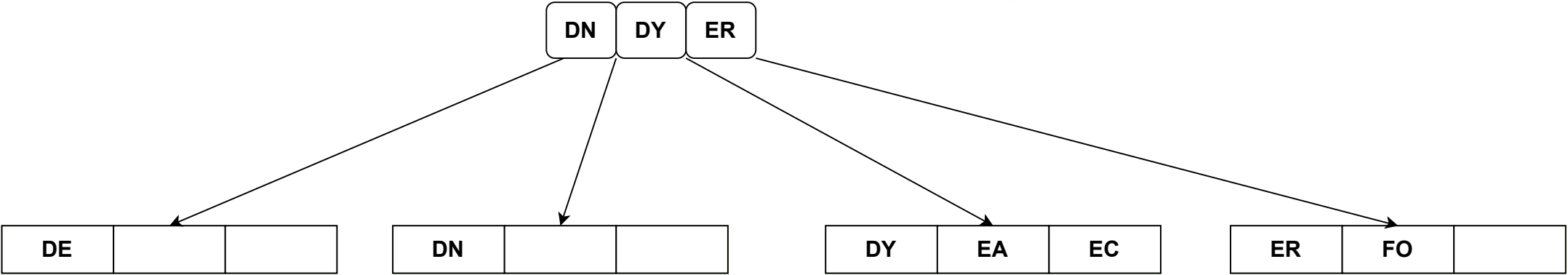




**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC*, EH, FB, FC, GK, GZ, GY

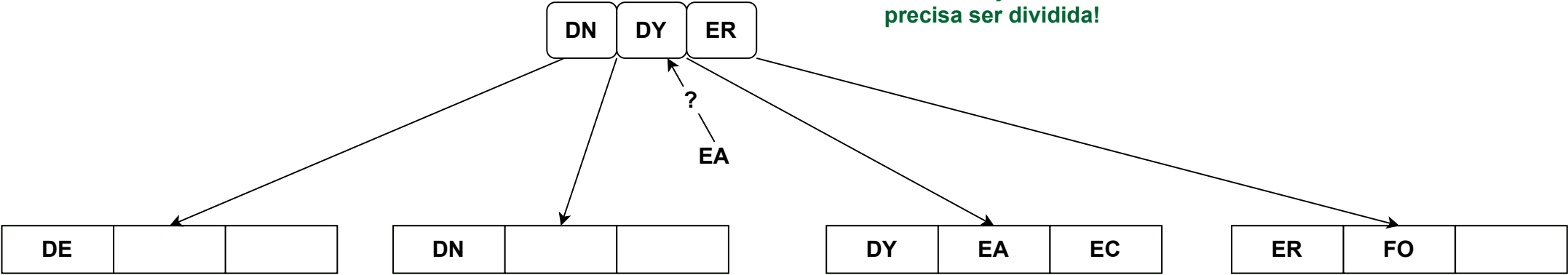
Nova divisão necessária, para  
inserir o EH,...



**Sequência de Inserção:**

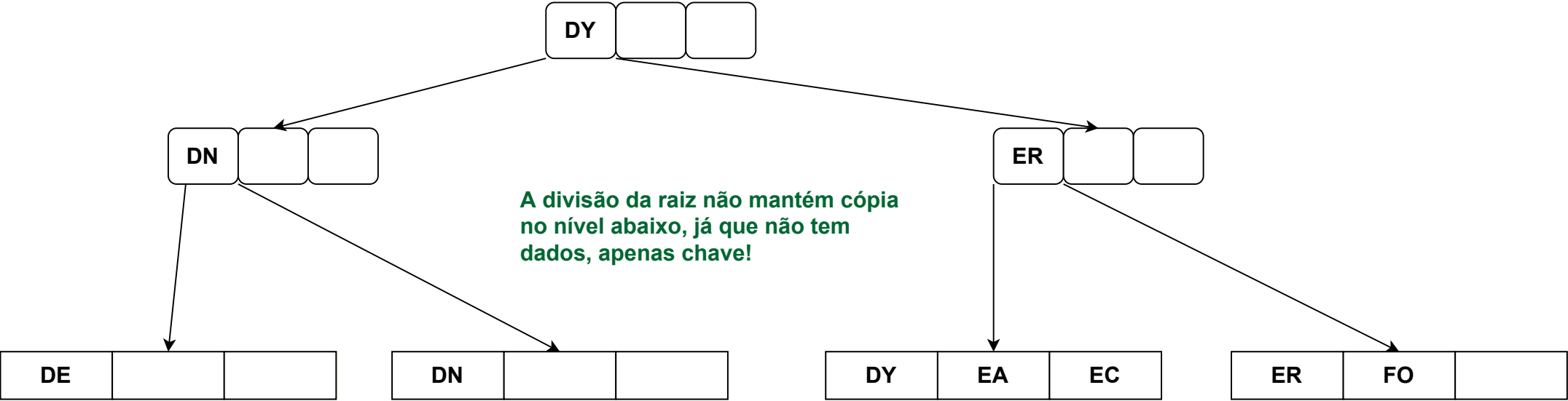
*DY, ER, FO, DE, DN, EA, EC*, EH, FB, FC, GK, GZ, GY

Nova divisão necessária, para  
inserir o EH,...  
Mas a raiz já está cheia e também  
precisa ser dividida!



**Sequência de Inserção:**

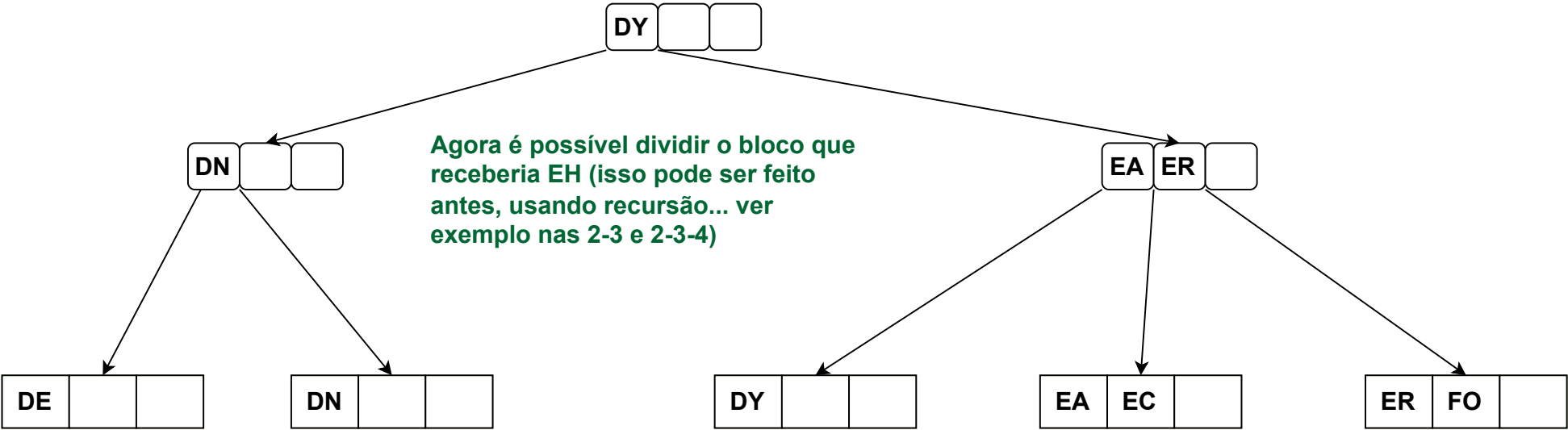
*DY, ER, FO, DE, DN, EA, EC*, EH, FB, FC, GK, GZ, GY



**Sequência de Inserção:**

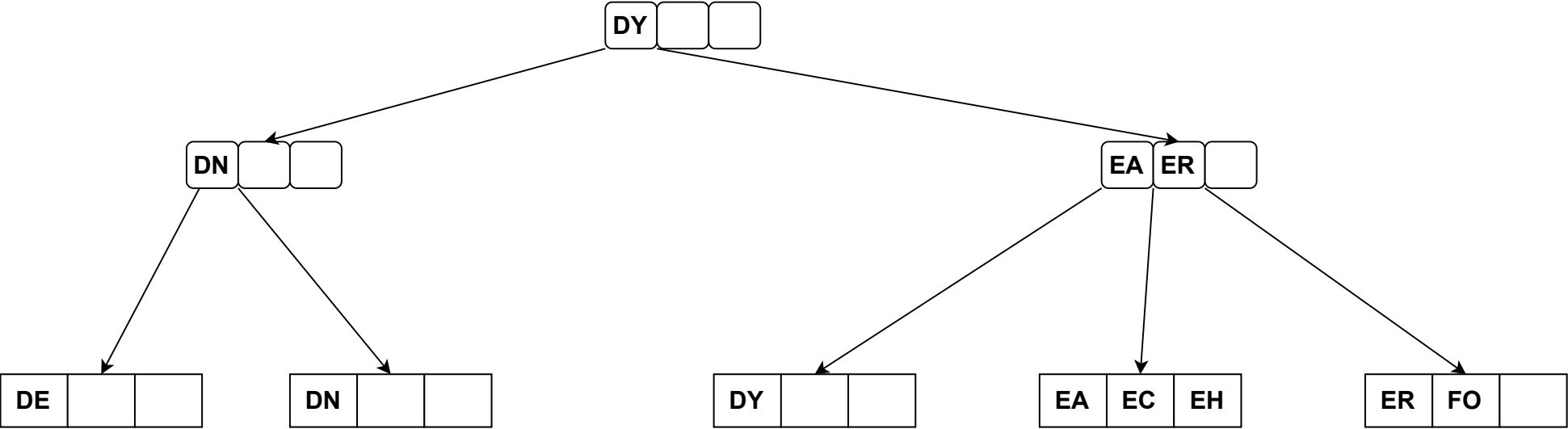
*DY, ER, FO, DE, DN, EA, EC*, EH, FB, FC, GK, GZ, GY

Agora é possível dividir o bloco que receberia EH (isso pode ser feito antes, usando recursão... ver exemplo nas 2-3 e 2-3-4)



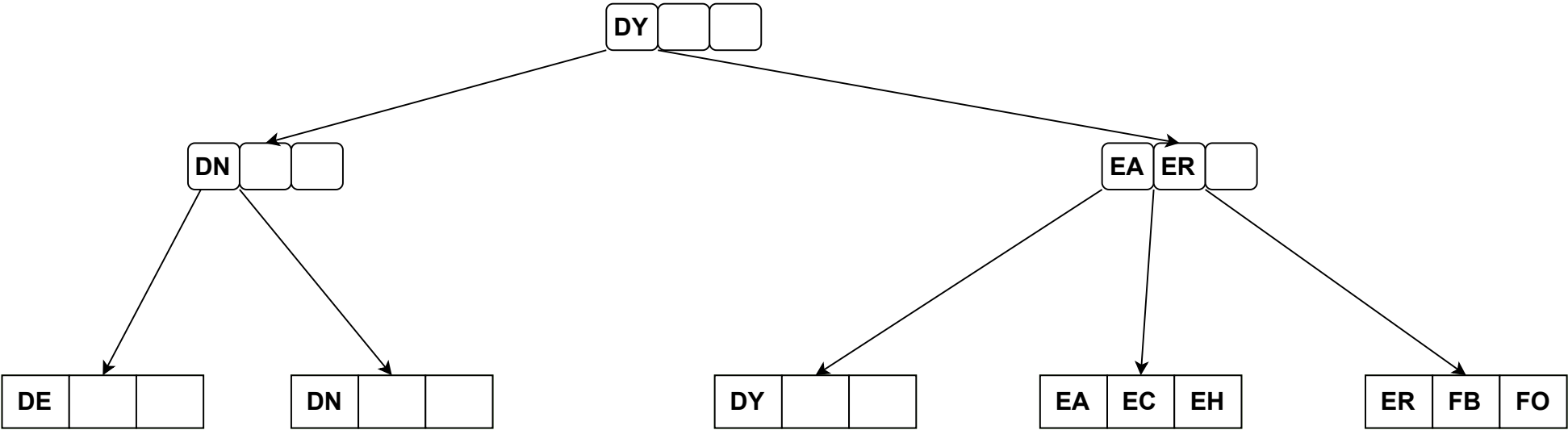
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH*, FB, FC, GK, GZ, GY



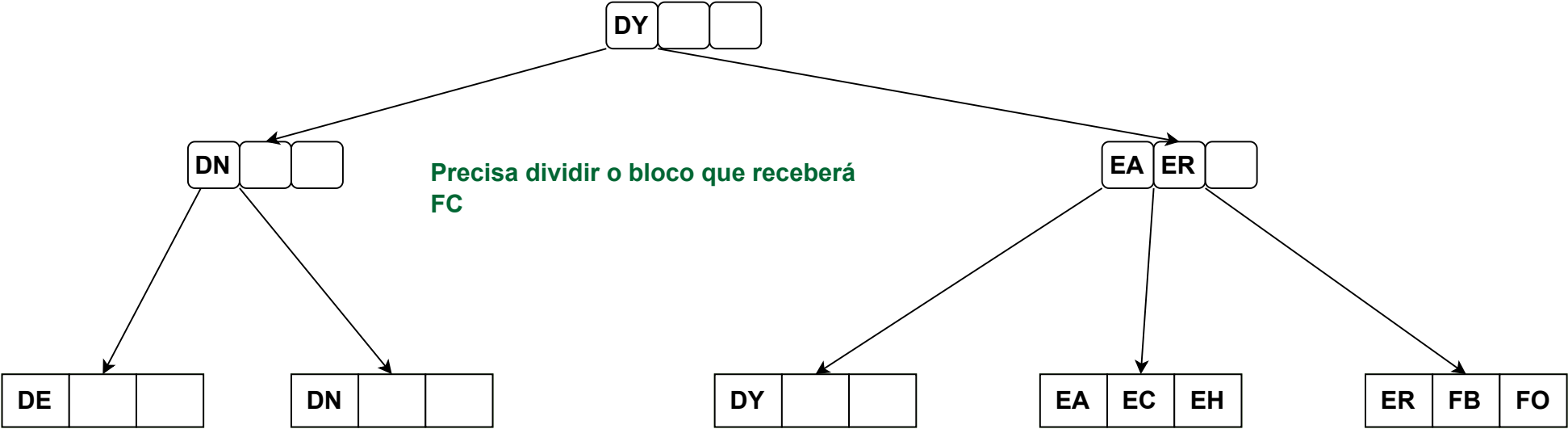
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB*, FC, GK, GZ, GY



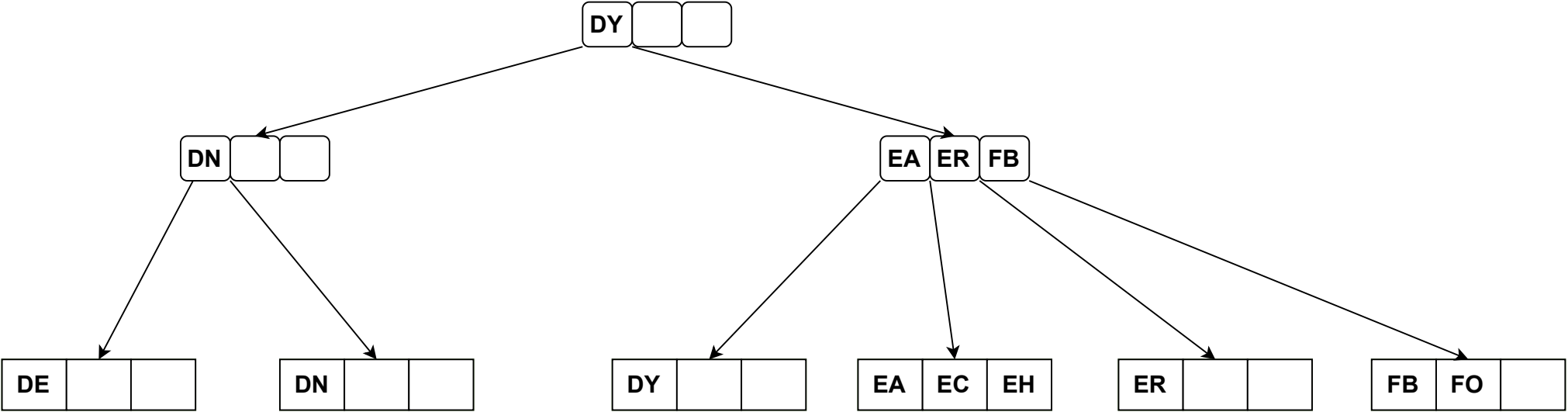
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB*, FC, GK, GZ, GY



**Sequência de Inserção:**

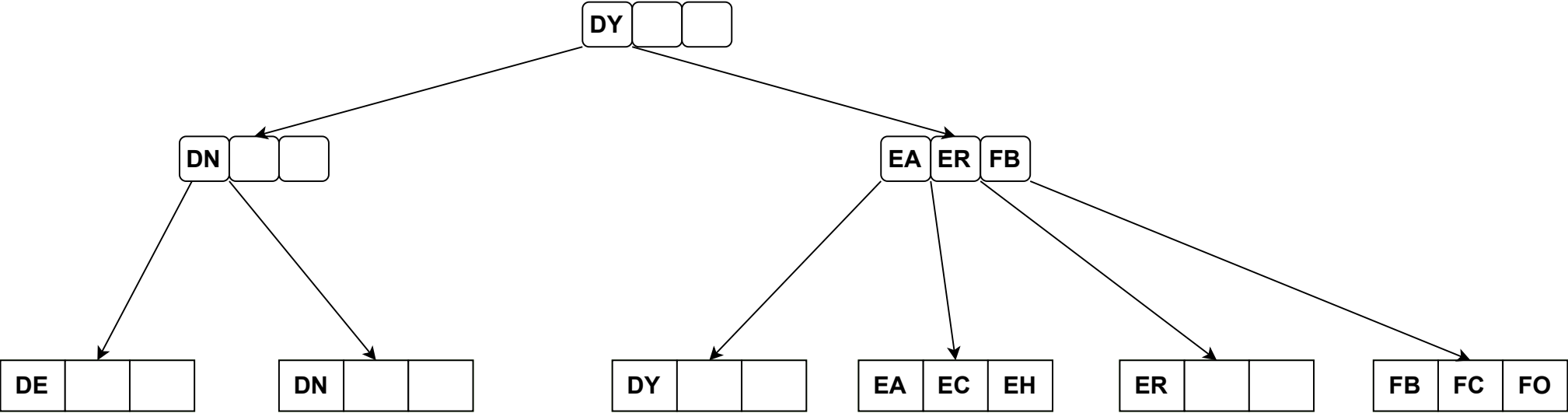
*DY, ER, FO, DE, DN, EA, EC, EH, FB*, FC, GK, GZ, GY





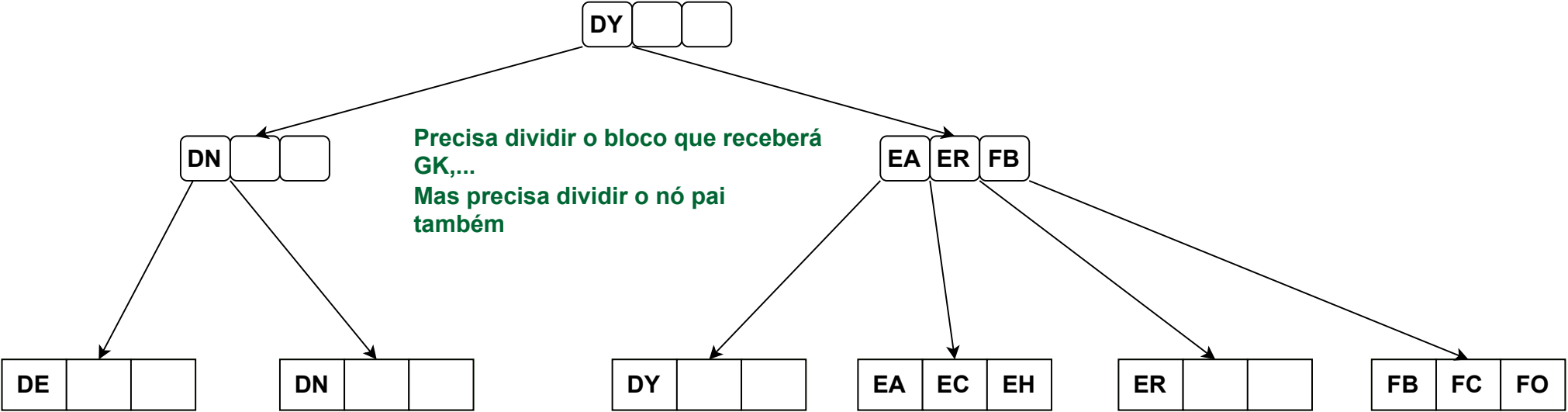
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC*, GK, GZ, GY



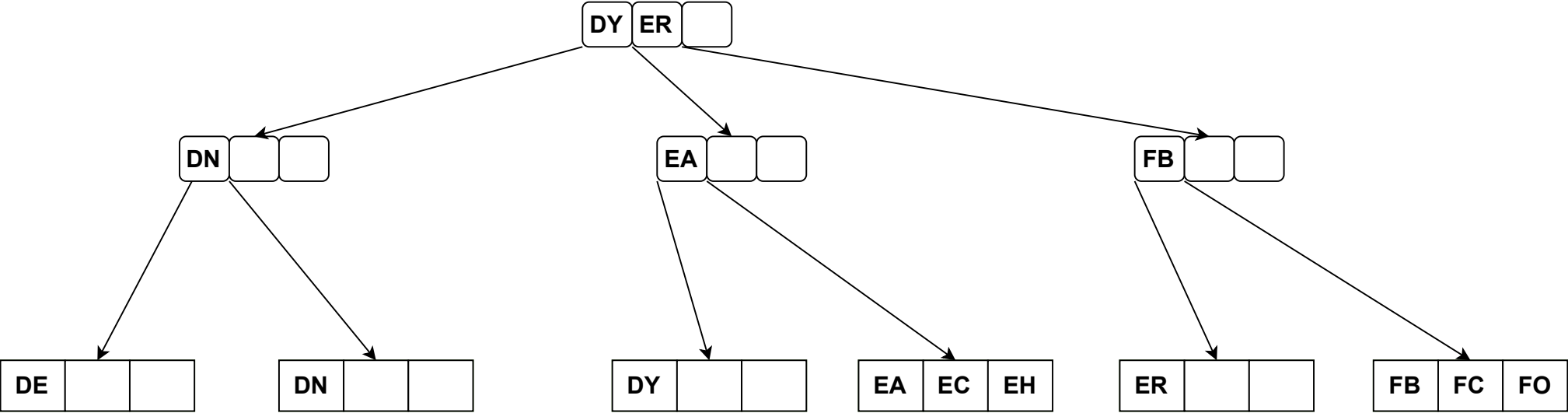
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC*, GK, GZ, GY



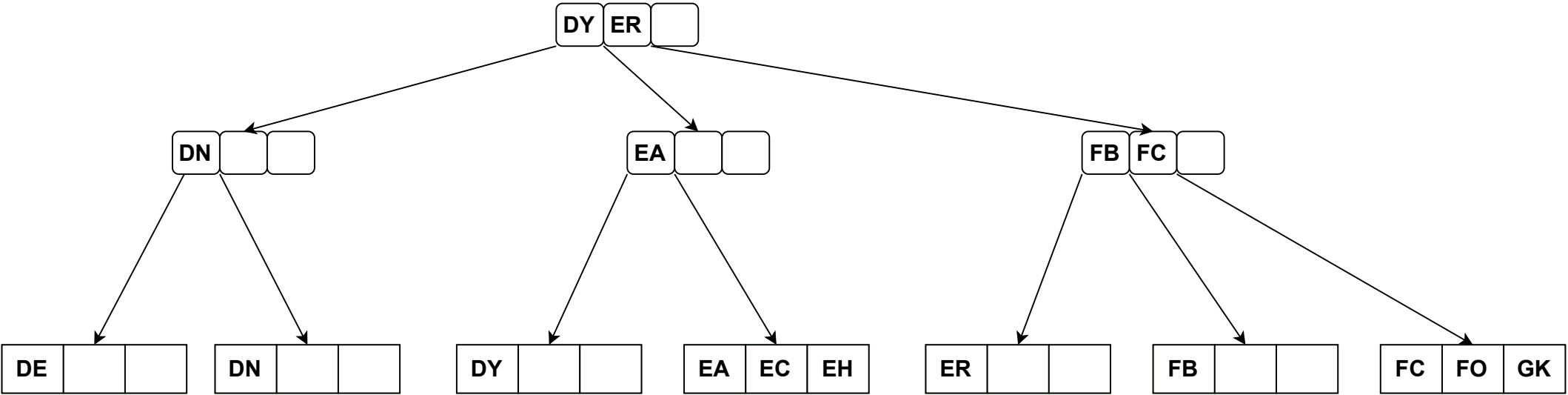
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC*, GK, GZ, GY



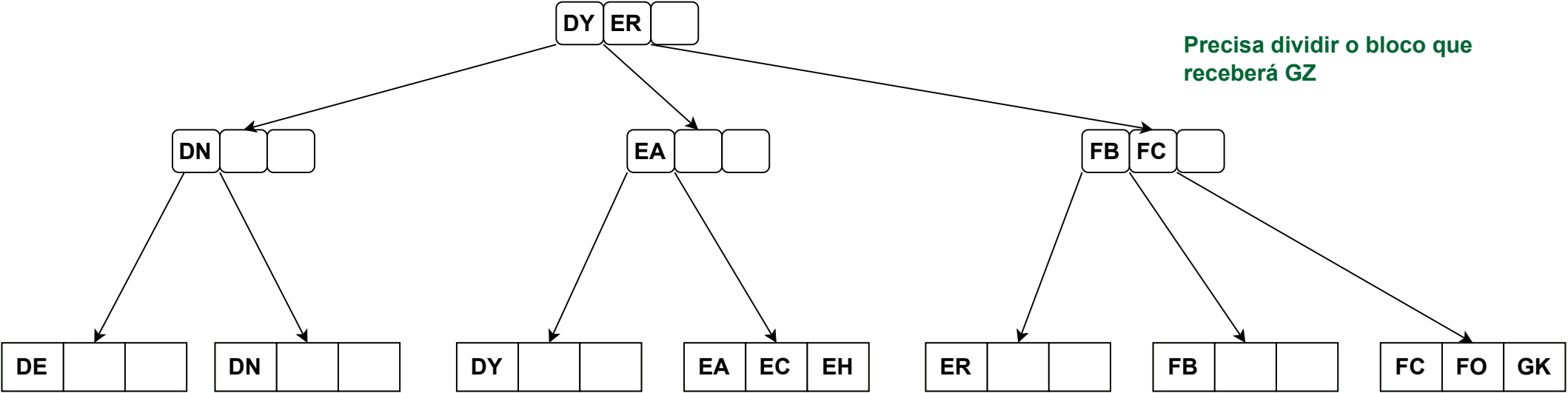
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK*, GZ, GY



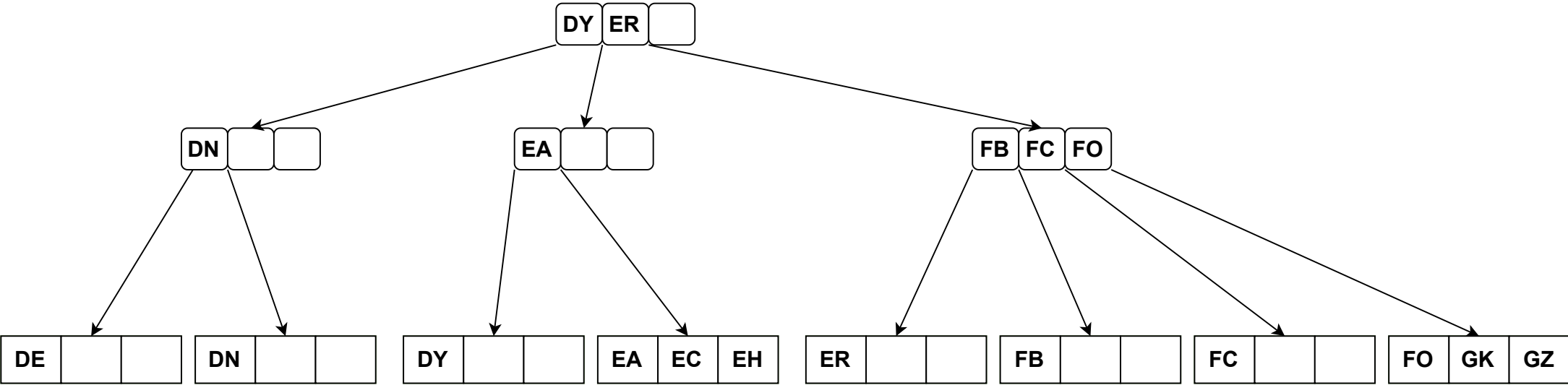
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK*, GZ, GY



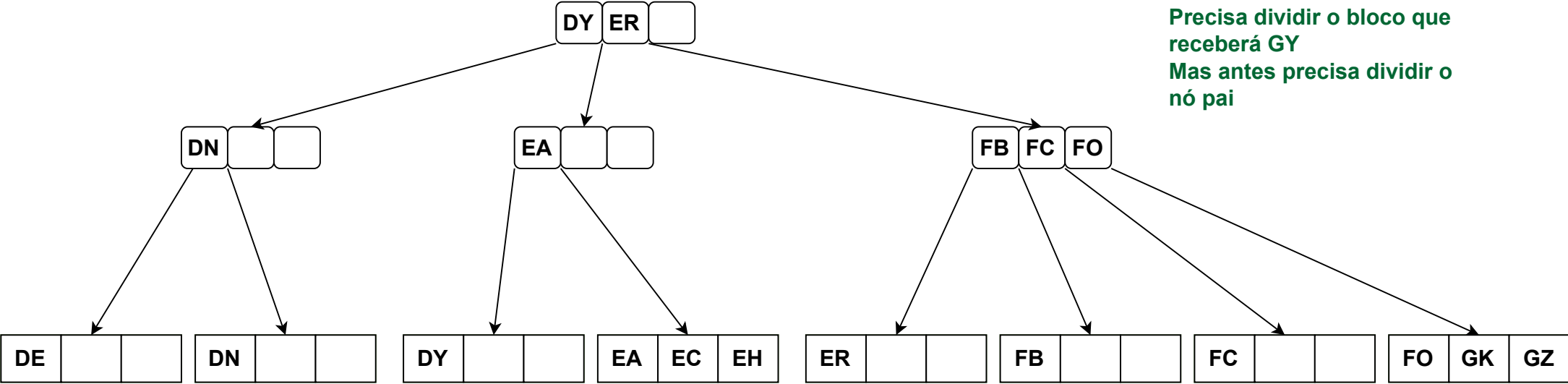
**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ*, GY



**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ*, GY

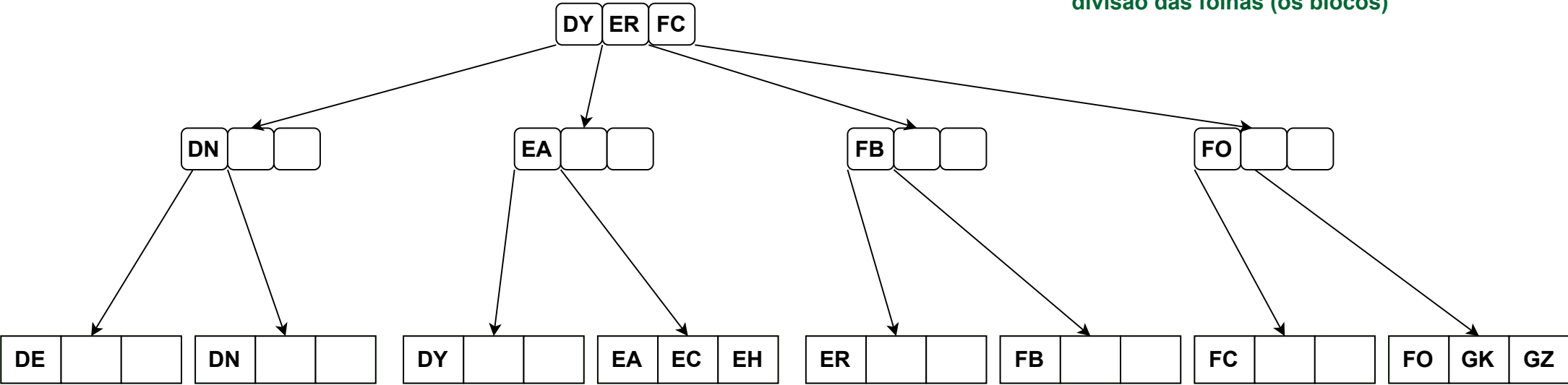


Precisa dividir o bloco que  
receberá GY  
Mas antes precisa dividir o  
nó pai

**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ*, GY

Perceba que divisão de nó folha não tem cópia da chave, ao contrário da divisão das folhas (os blocos)

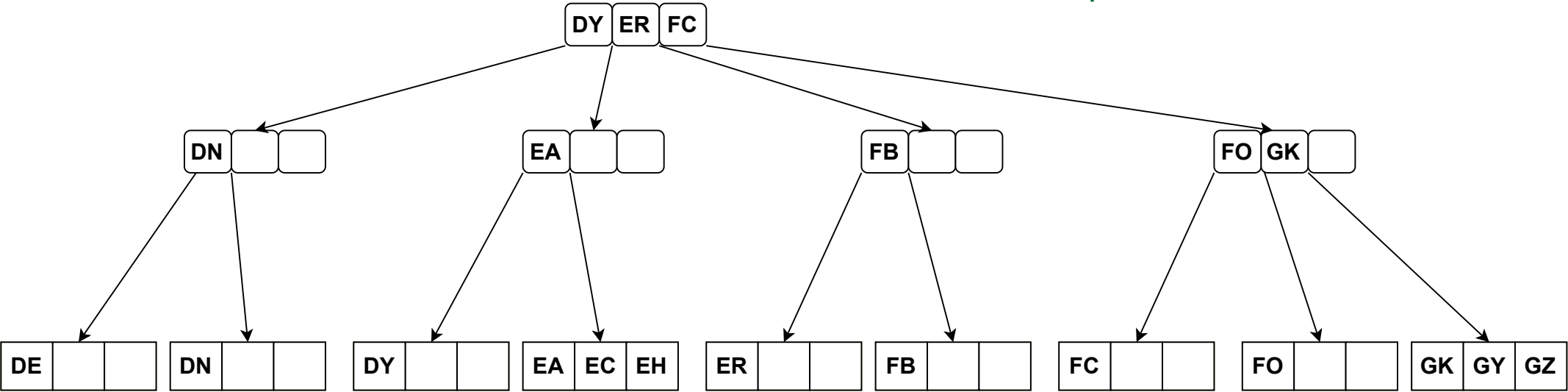




**Sequência de Inserção:**

*DY, ER, FO, DE, DN, EA, EC, EH, FB, FC, GK, GZ, GY*

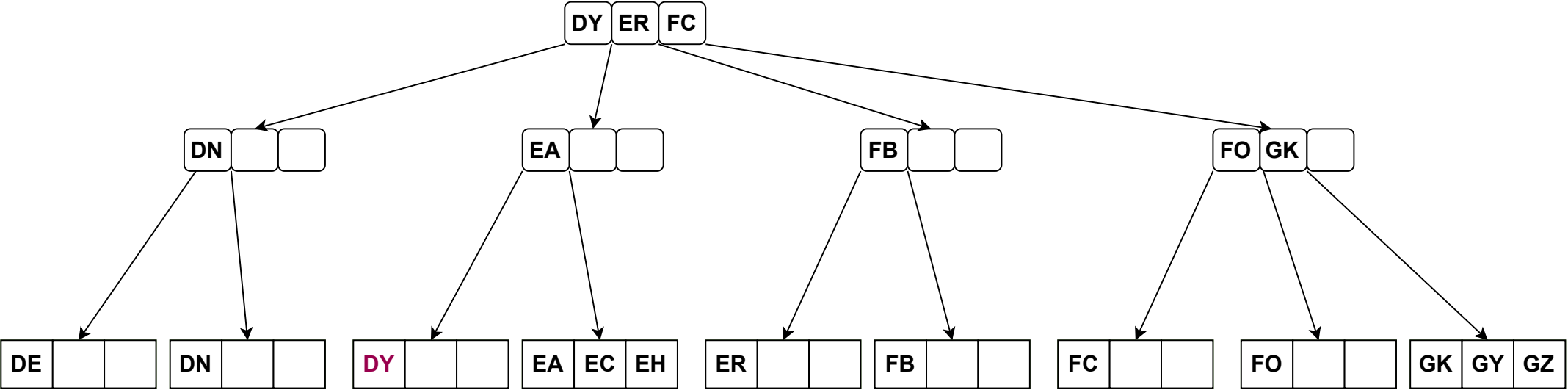
Aqui tivemos cópia da chave na divisão,  
pois GK estava em nó folha.



**Sequência de Remoção:**

DY, FO, FC, EC, EH

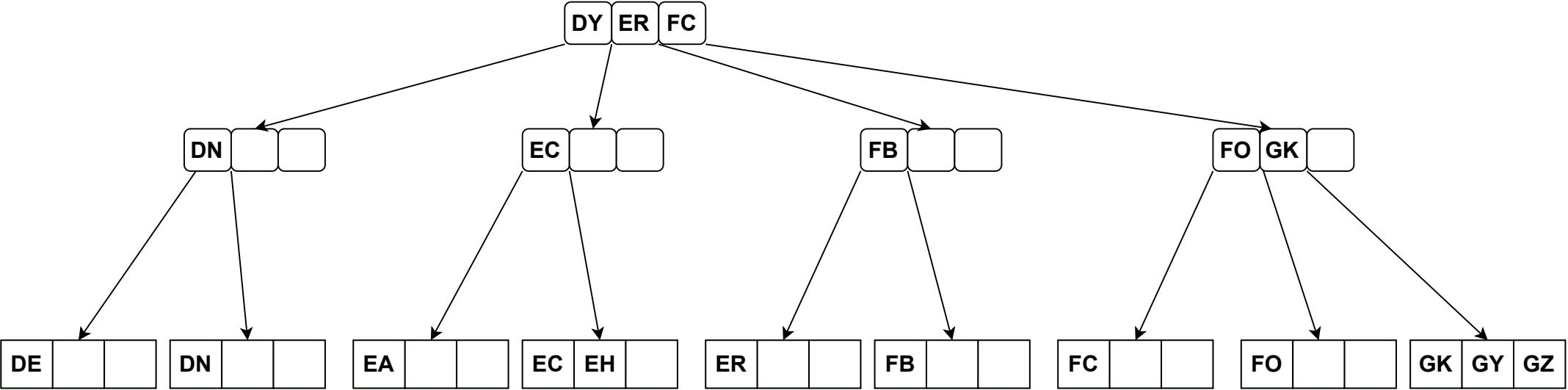
A remoção do DY deixará o bloco com menos elementos que o requerido, mas é possível fazer rotação com o bloco irmão. Nesse caso, iremos passar EA para o nó com o elemento excluído...



**Sequência de Remoção:**

**DY**, FO, FC, EC, EH

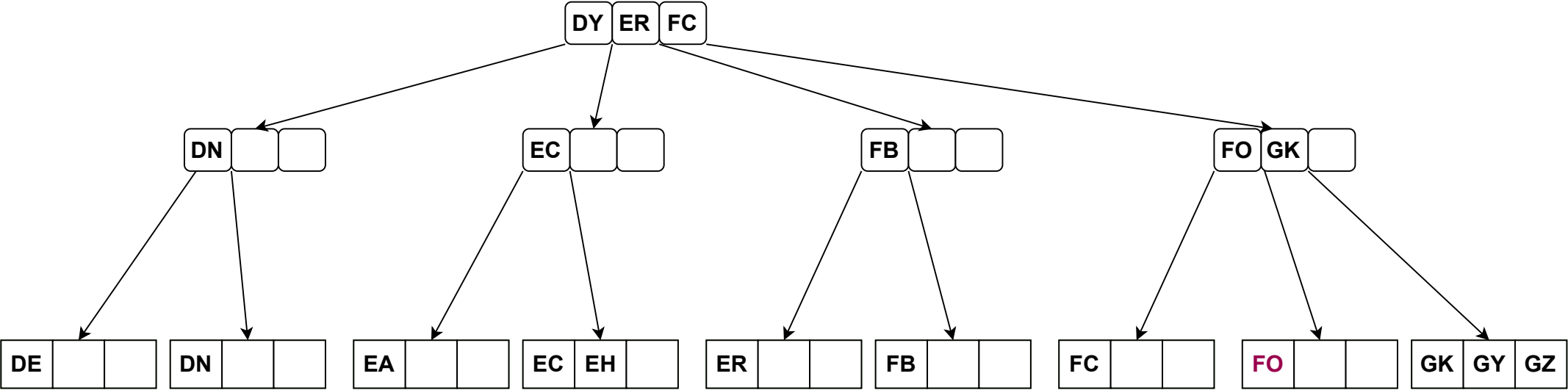
Com a rotação, note que a chave também foi alterada (sofreu rotação), para que a árvore de índices continue válida.



**Sequência de Remoção:**

**DY**, FO, FC, EC, EH

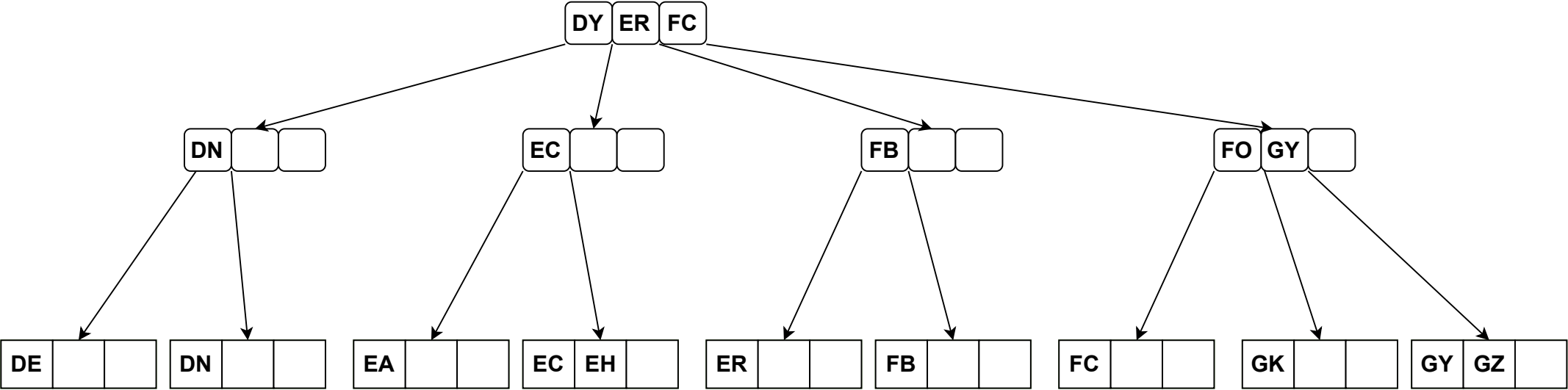
Com FO, adotamos a mesma abordagem: fazemos rotação com o irmão... Notem que o irmão à esquerda não tem elementos suficientes, o bloco pode fazer rotação com qualquer irmão, geralmente consultando primeiro o bloco à esquerda e depois o bloco à direita.



**Sequência de Remoção:**

**DY, FO**, FC, EC, EH

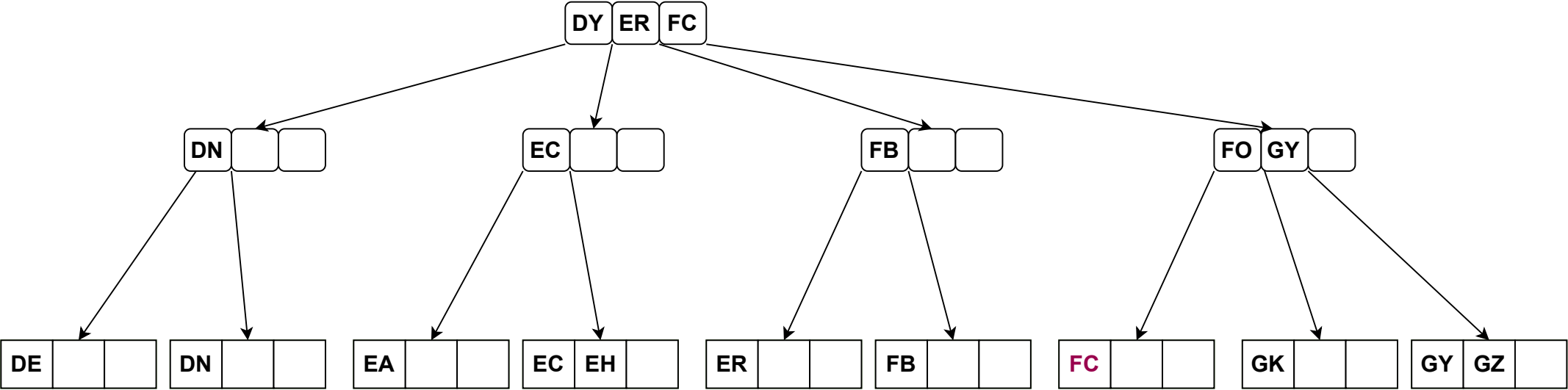
Percebam que o FO continua como chave, mesmo não tendo um valor no bloco. Isso não importa, pois a árvore continua sendo válida para buscas.



A remoção de FC é mais complicada um pouco, já que ele não tem um irmão à esquerda ou direita com o qual possa fazer rotação... Nesse caso, ele se funde com um irmão (à esquerda ou direita). Geralmente ele tenta primeiro com irmão à esquerda (que FC não tem) para depois com irmão à direita.

**Sequência de Remoção:**

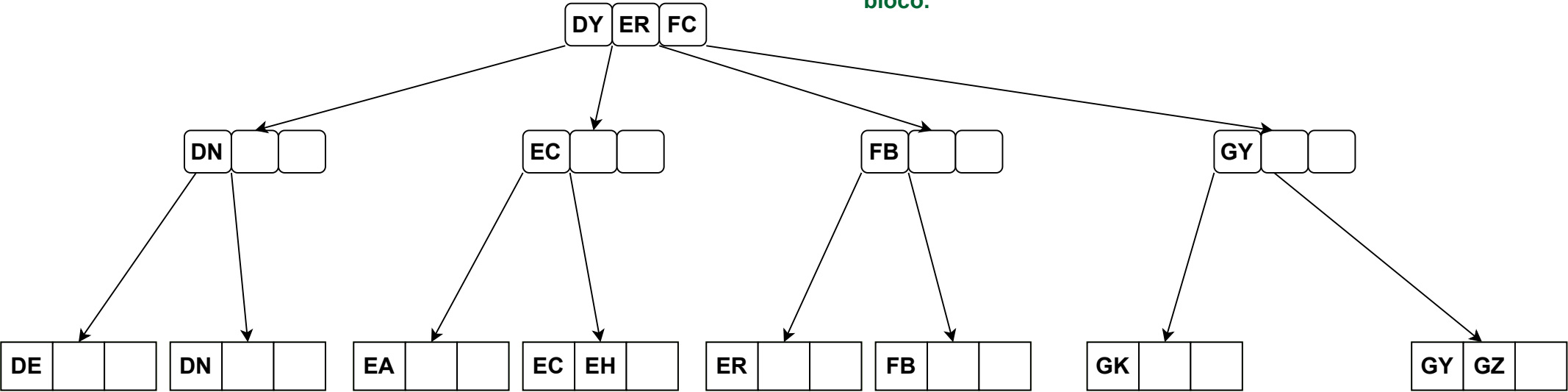
**DY, FO**, FC, EC, EH



**Sequência de Remoção:**

**DY, FO, FC**, EC, EH

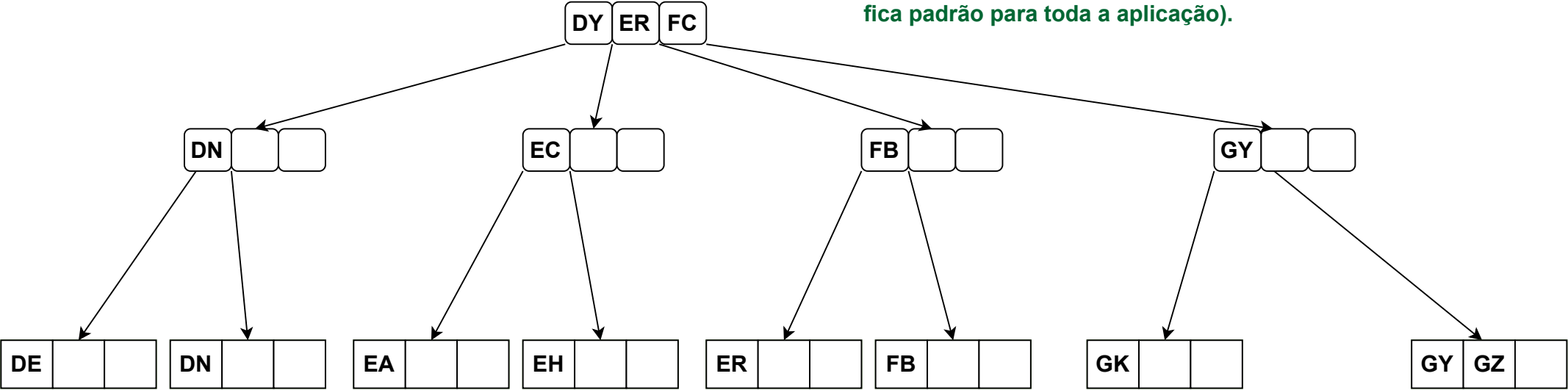
Com a fusão, dois blocos se unem, e a chave que existia é removida. Na árvore B, a chave descia ao nó. Com a B+, o processo é mais simples: basta remover e ajustar os nós, não importando se há um dado com aquela chave no bloco.



**Sequência de Remoção:**

*DY, FO, FC, EC*, EH

A remoção de EC é muito tranquila, mas EH vai implicar em fusão no nó pai (que fica com um filho apenas). Como o nó pai tem irmão à esquerda e à direita, poderia se fundir com qualquer um dos dois. Em geral, escolhe-se fusão à esquerda (mas isso depende da implementação e fica padrão para toda a aplicação).





**Sequência de Remoção:**

*DY, FO, FC, EC, EH*

Na fusão, a chave que separava os dois irmão (DY) substituiu a chave anterior EC, para garantir a integridade da árvore.

Vejam que na árvore de índices existem chaves (FC e DY) que não estão nos blocos, são usadas apenas para busca.

