

Árvores B

IFRN/RN

Árvores B

- Criadas por Bayes e McCreight em 1972
- Em algumas aplicações a quantidade de dados é muito grande para que as chaves possam ser armazenadas somente em memória
- É necessário o uso de memória secundária, o que ocasiona um gasto de tempo significativo para acesso a um só nó dos dados

Árvores B

- Utiliza mais de uma chave em cada nó da estrutura
- Assegura que as folhas estão sempre no mesmo nível
- São utilizadas como forma de armazenamento em memória secundária

Árvores B

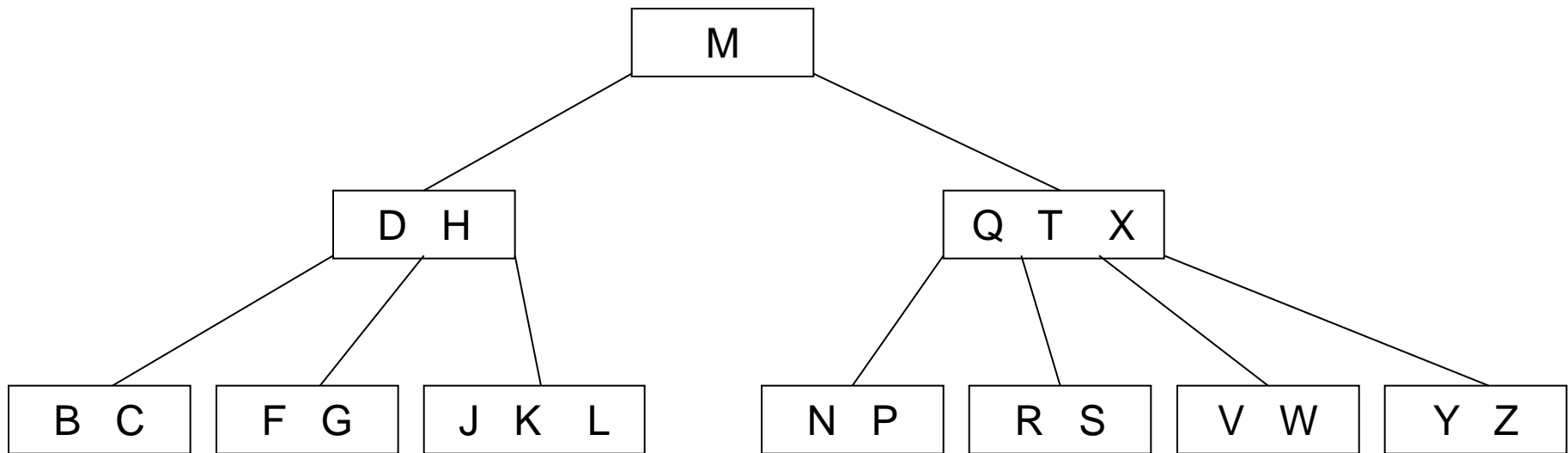
- Definição:
 - Seja t a ordem da árvore B. Uma árvore B de ordem t é uma árvore ordenada que é vazia, ou que satisfaz as seguintes condições:
 1. A raiz é uma folha ou tem no mínimo dois **filhos**;
 2. Cada nó diferente do raiz e das folhas possui no mínimo t **filhos**;
 3. Cada nó tem no máximo $2t$ **filhos**;
 4. Todas as folhas estão no mesmo nível

Árvores B

- Como consequência da definição:
 - Cada nó (página) possui entre $t-1$ e $2t-1$ **chaves**, exceto o raiz que possui entre 1 e $2t-1$ **chaves**

Árvores B

- Exemplo $t=3$,
 - $(t-1$ e $2t-1$ chaves) e $(t$ e $2t$ filhos)

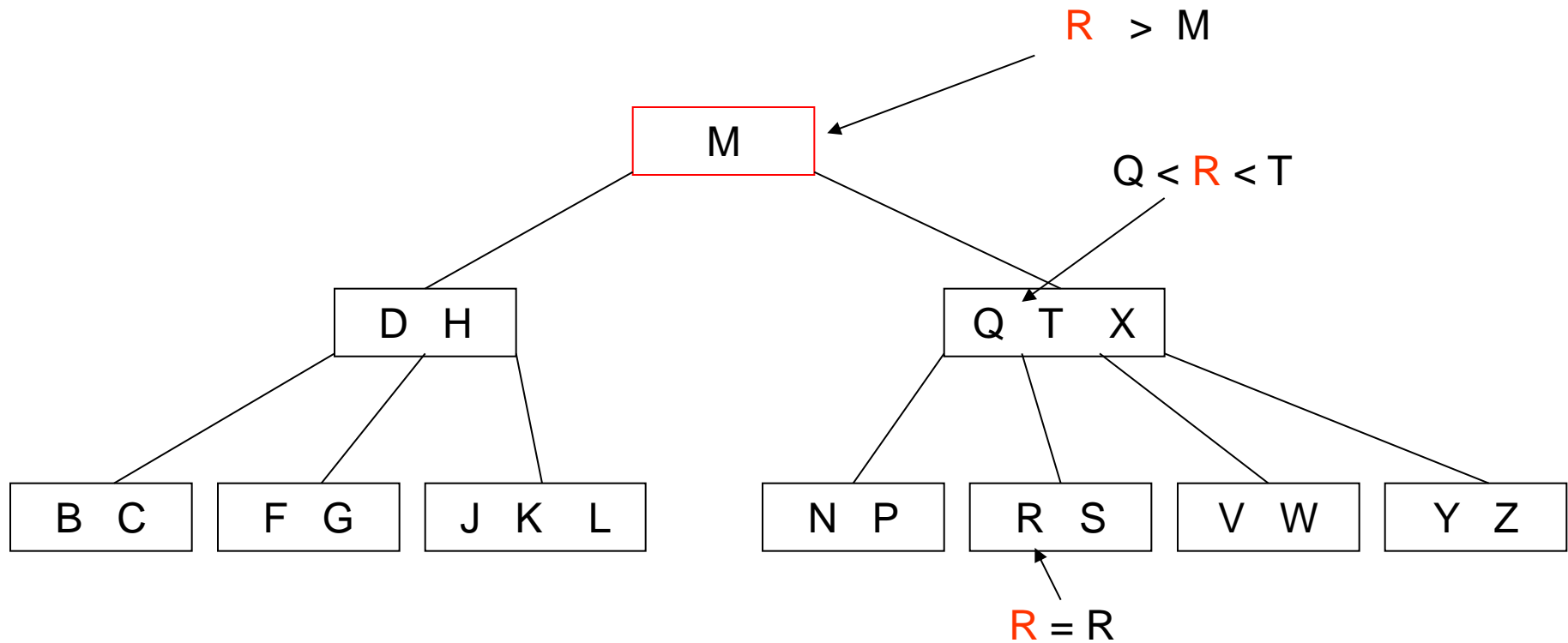


Árvores B

- Busca
 - Compara a chave x com a(s) chave(s) do nó raiz. Se a chave não estiver no raiz, a busca deve prosseguir em um certo filho dessa página.
 - Escolhe-se qual filho explorar de forma parecida com a pesquisa realizada na árvore binária de pesquisa

Árvores B

- Busca a chave R

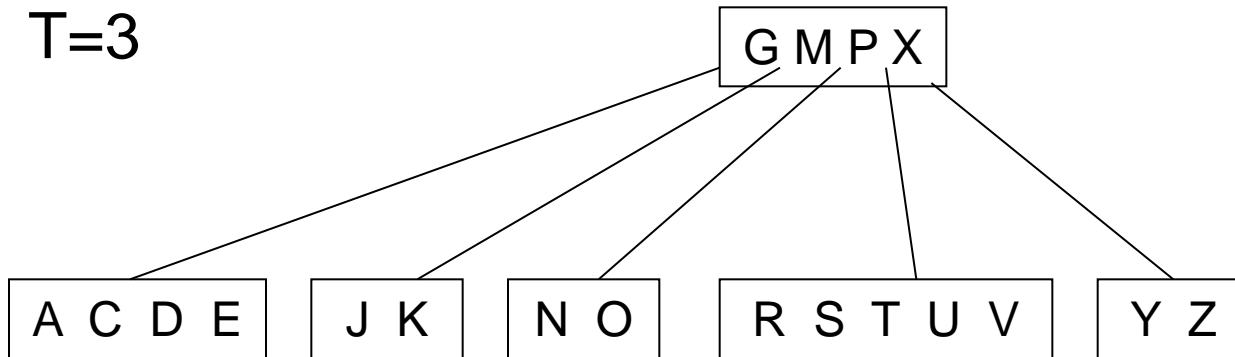


Árvores B

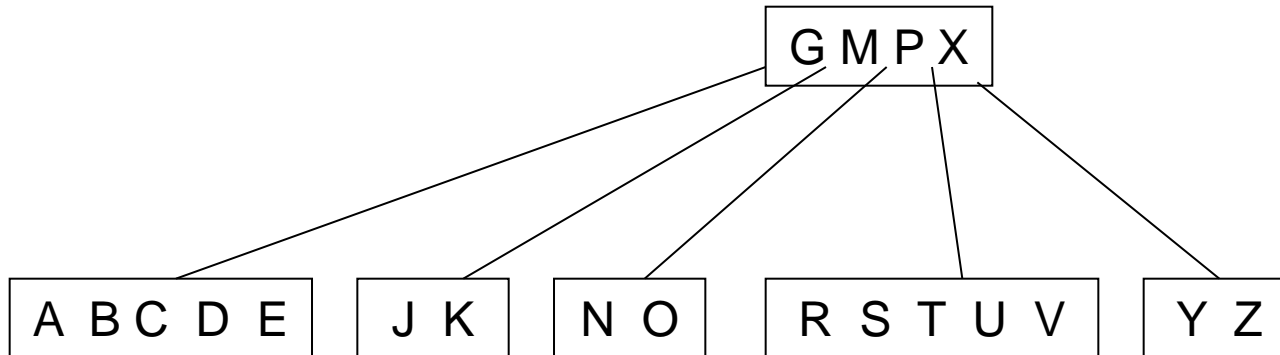
- Inserir
 - Passos:
 - Usar o procedimento de busca
 - Se a inserção é válida, incluir a nova entrada na posição da folha apontada pela busca
 - Verificar se a página necessita de cisão (nó com mais de $2t-1$ **chaves** ou raiz com $2t-1$ **chaves** – limite)

Árvores B

T=3

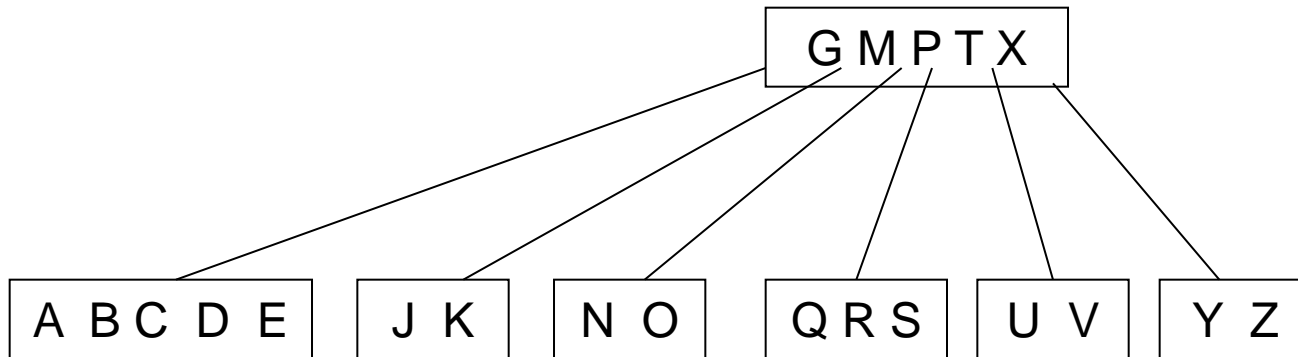
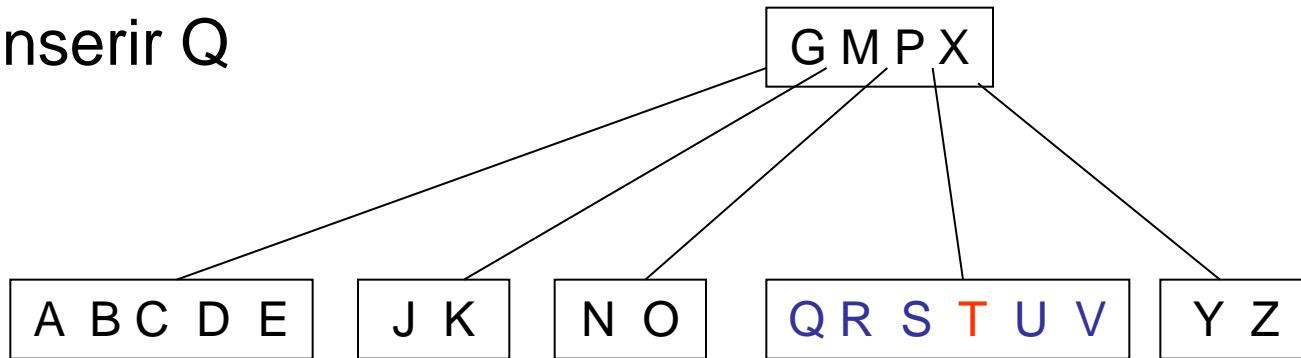


Inserir B



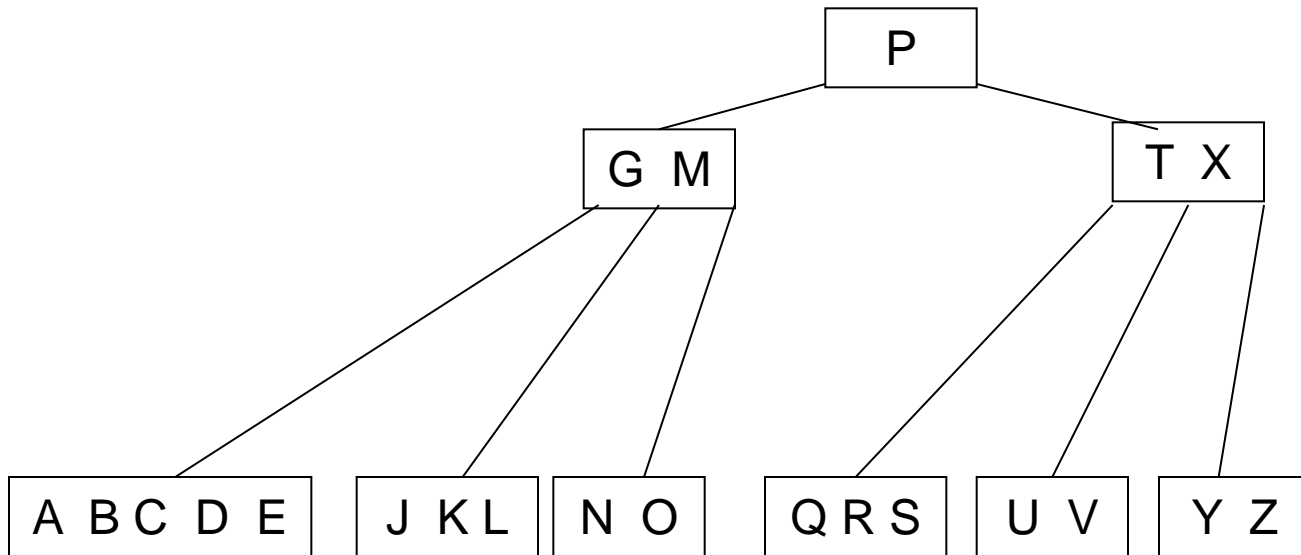
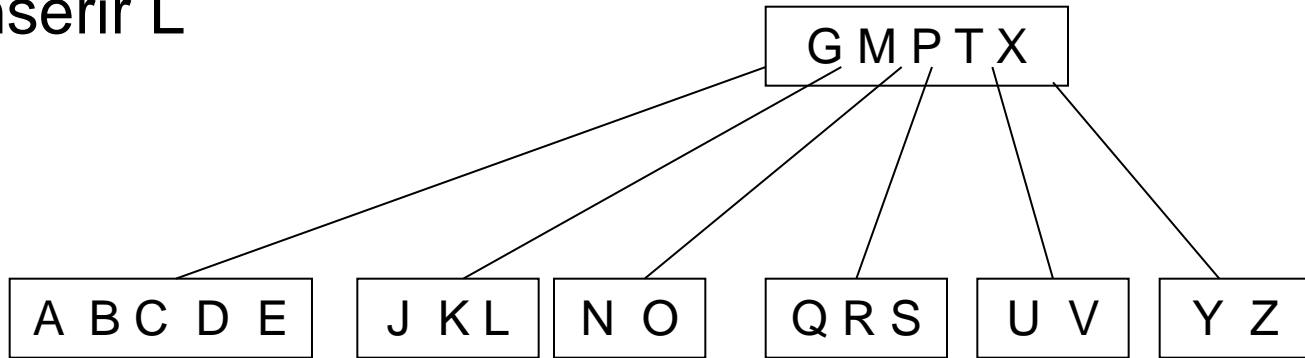
Árvores B

Inserir Q



Árvores B

Inserir L



Árvore B

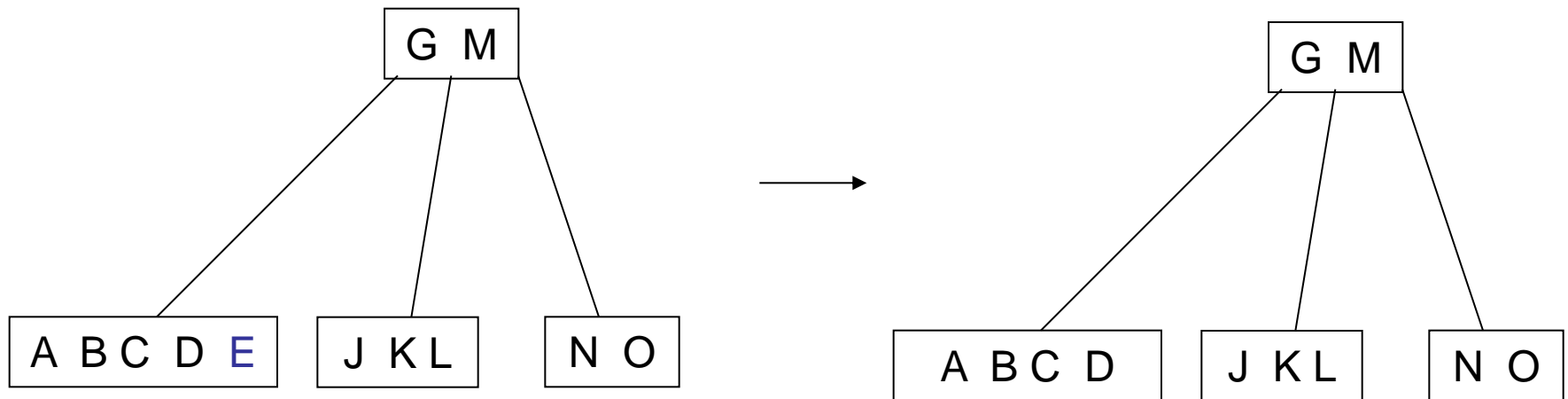
- Remover
 - Passos:
 - Usar o procedimento de busca
 - Segue com um dos três casos abaixo:

Árvore B

- Remove

1) Se a chave x está no nó x e x é folha, elimine x do nó

Remove **E**



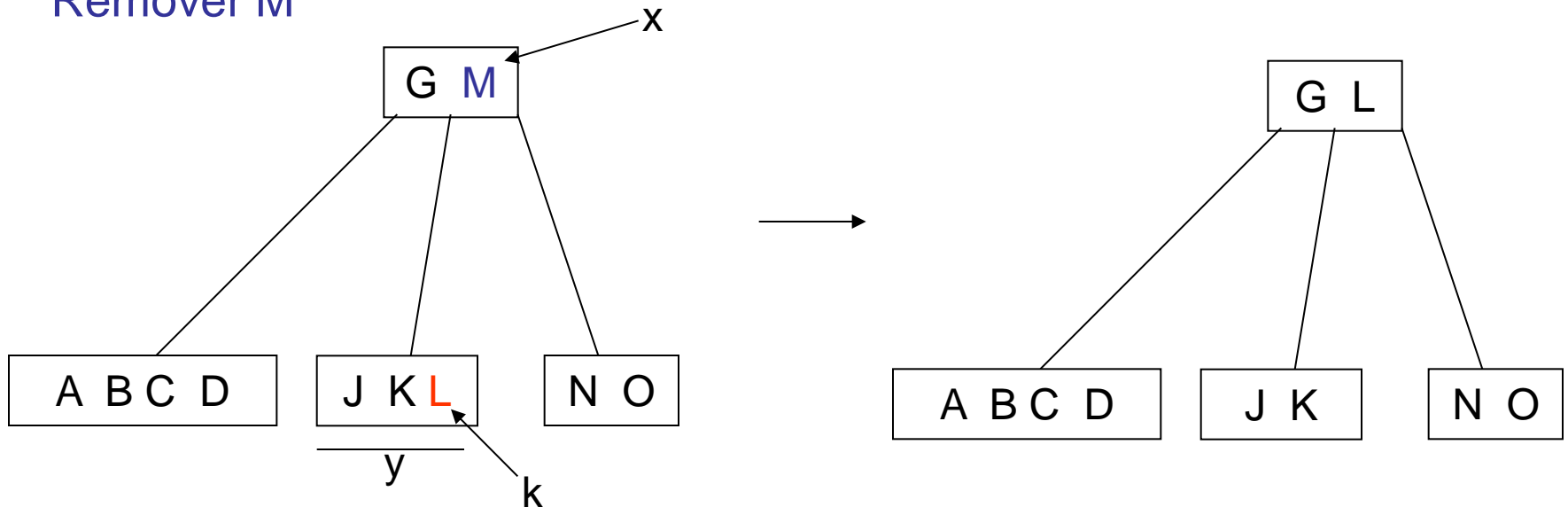
Árvore B

- Remove

2) Se a chave x está no nó s e s é interno, faça:

- a) Se o filho y anterior a x no nó s tem pelo menos t chaves (uma a mais que o mínimo), remova x e coloque a chave k antecessor de x

Remove M



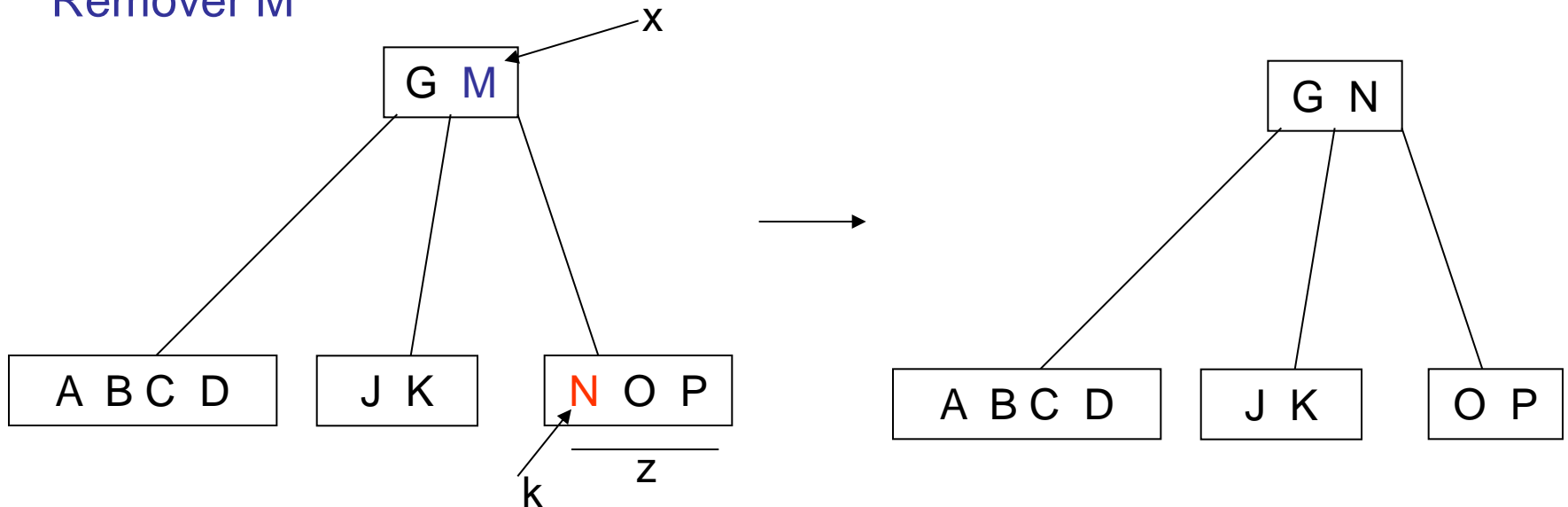
Árvore B

- Remove

2) Se a chave x está no nó s e s é interno, faça:

b) Se o filho z posterior a x no nó s tem pelo menos t chaves (uma a mais que o mínimo), remova x e coloque a chave k sucessor de x

Remove M



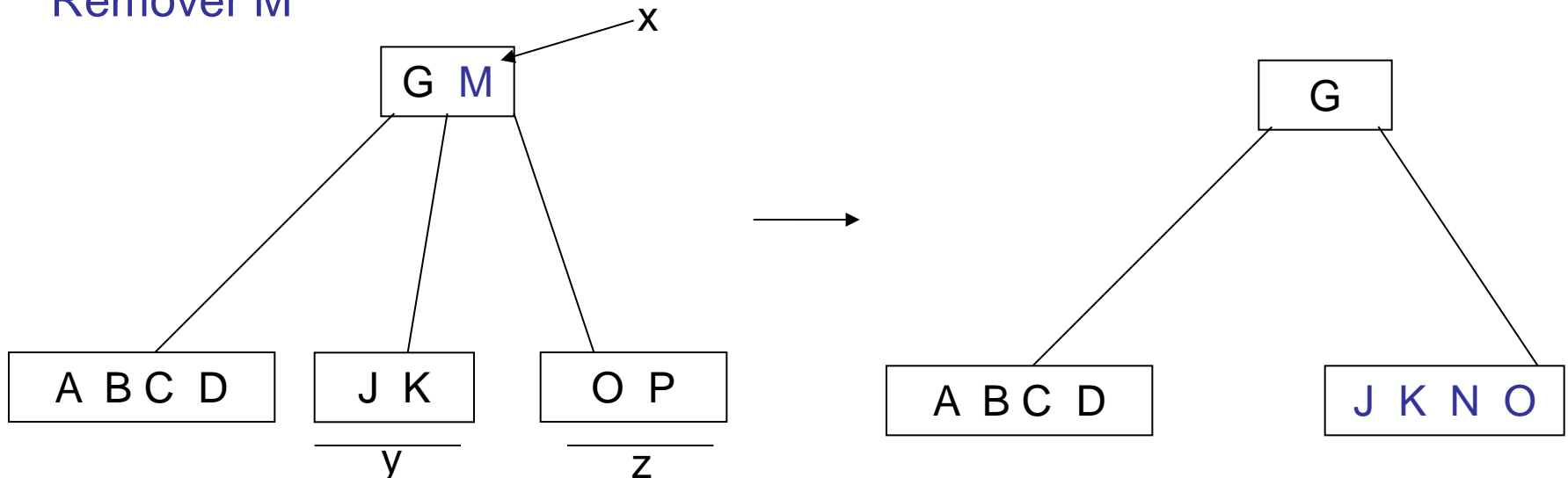
Árvore B

- Remove

2) Se a chave x está no nó s e s é interno, faça:

c) Se os filhos y e z a x no nó s tem pelo menos $t-1$ chaves cada um, remova x e faça uma fusão nos nós y e z

Remove M



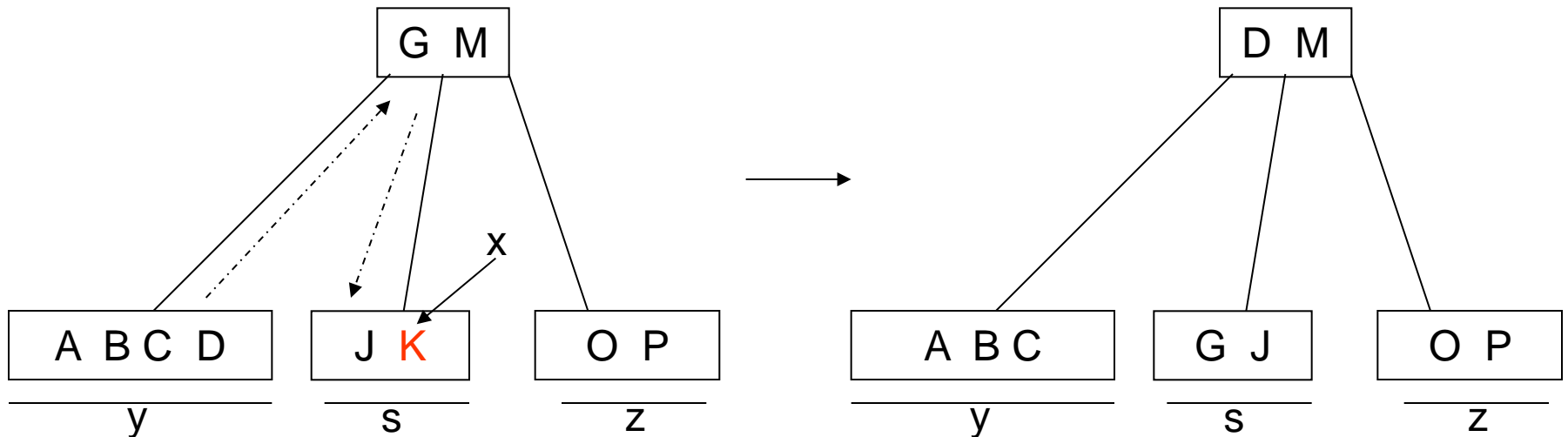
Árvore B

- Remover

3) Se a chave x está no nó s e s é externo com $t-1$ chaves, faça:

a) Se o nó s tem $t-1$ chaves e o irmão y ou z tem no mínimo t chaves, remova x de s pegue a chave emprestado de y ou z

Remover K



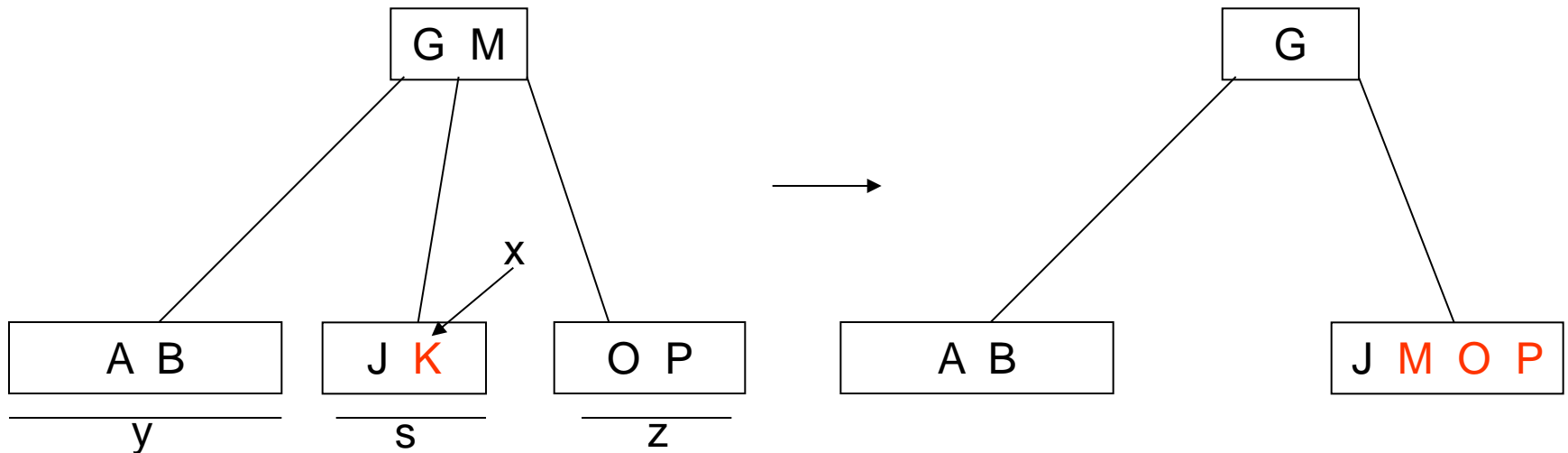
Árvore B

- Remover

3) Se a chave x está no nó s e s é externo com $t-1$ chaves, faça:

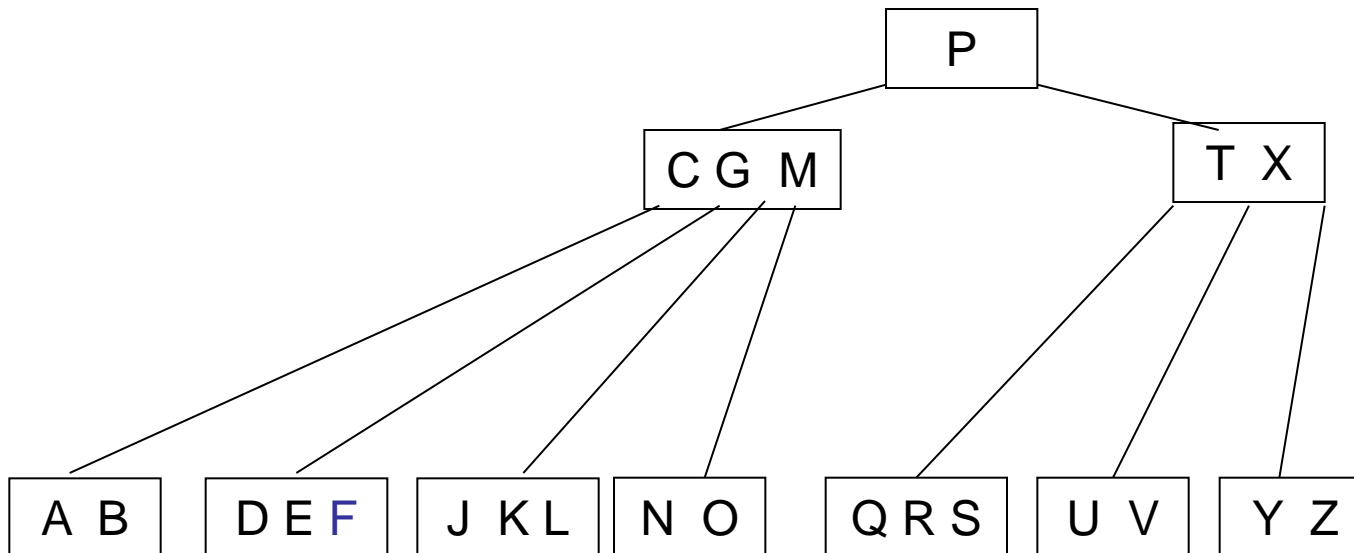
b) Se o nó s tem $t-1$ chaves e o irmão y ou z tem também $t-1$ chaves, remova x e faça uma fusão com y ou z

Remover K



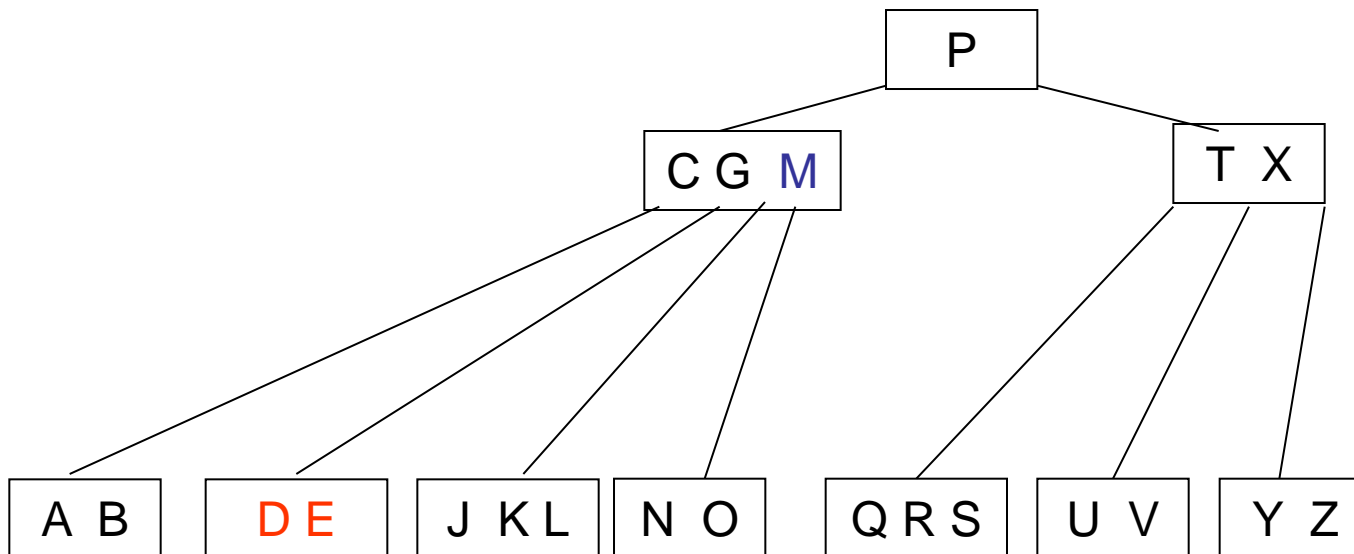
Árvore B

- Exemplo remoção, $t=3$
 - Remover F



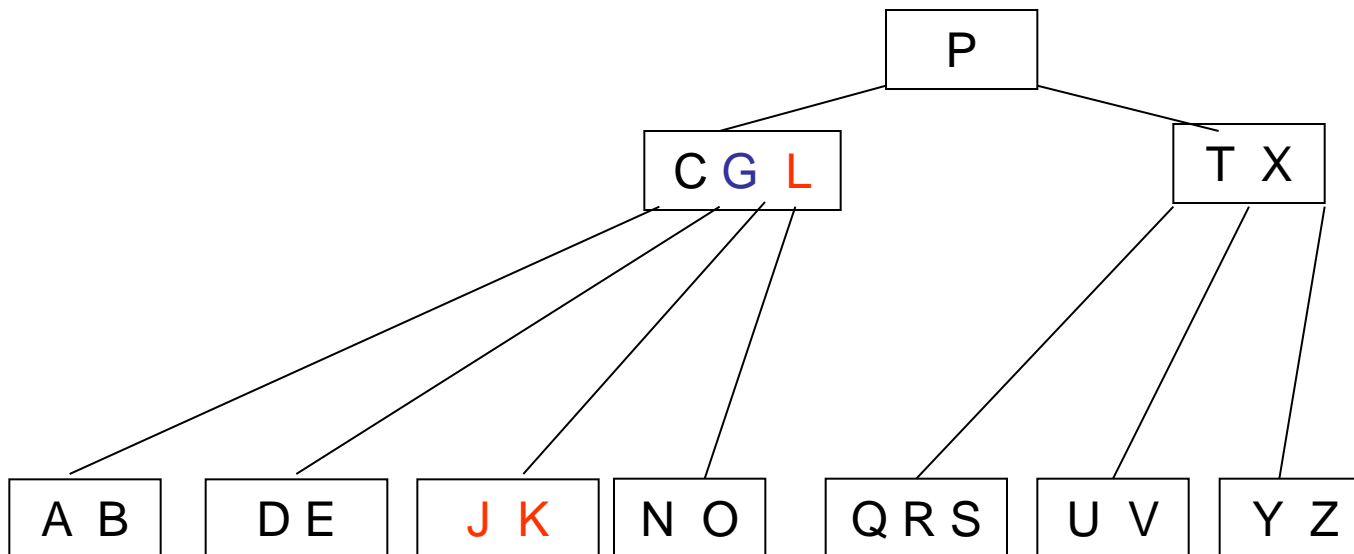
Árvore B

- Exemplo remoção
 - Remover M



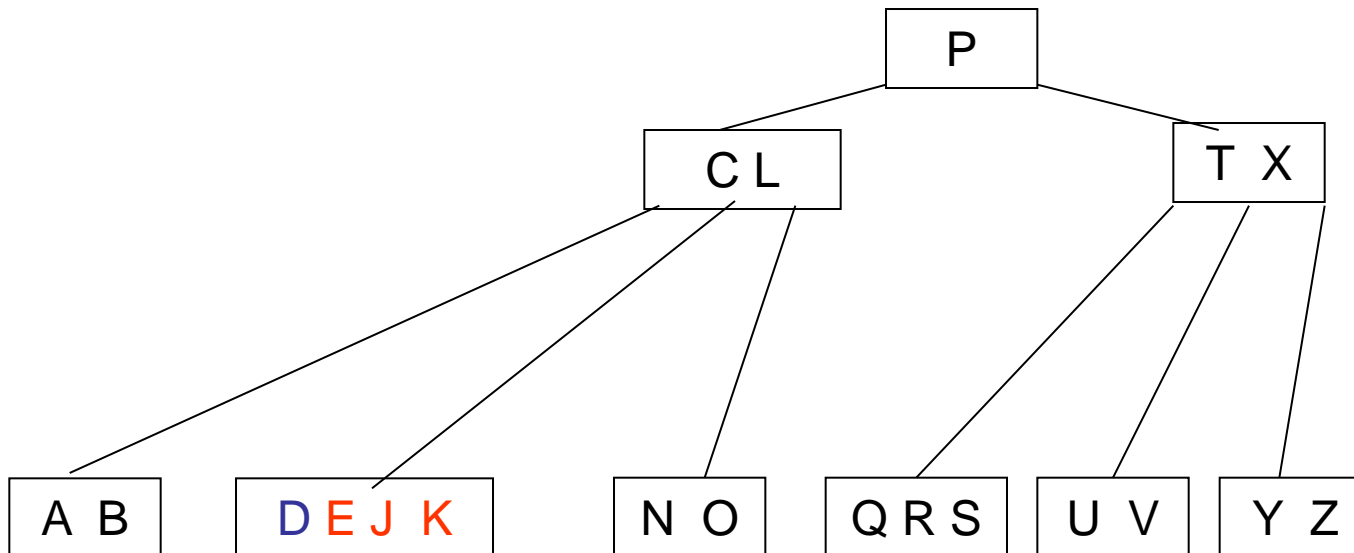
Árvore B

- Exemplo remoção
 - Remover G



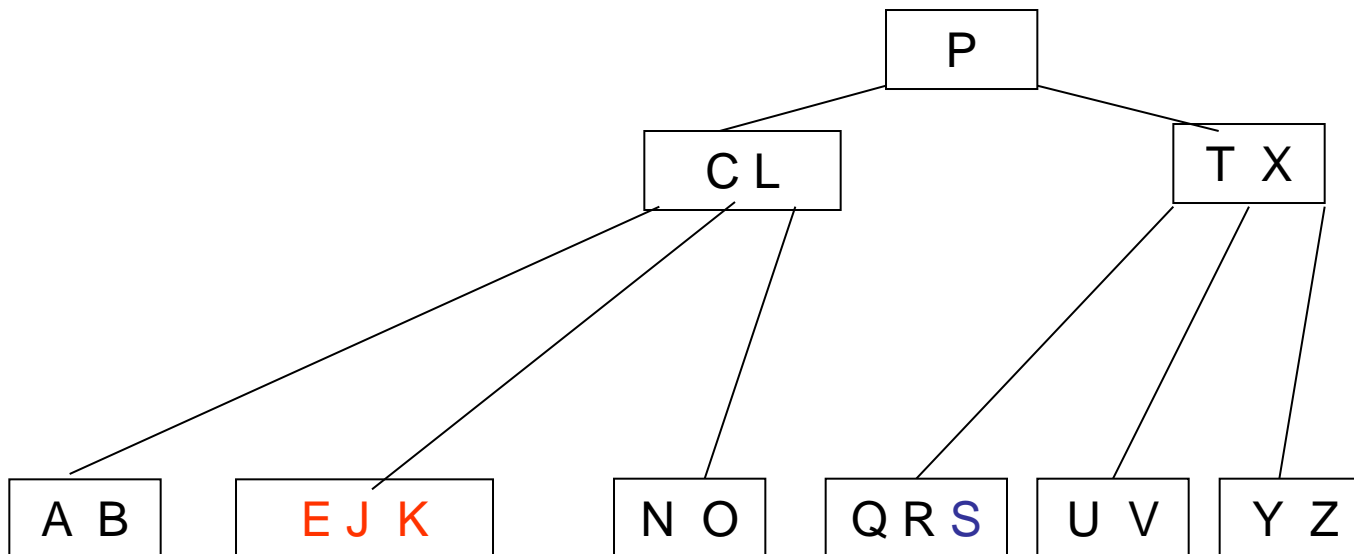
Árvore B

- Exemplo remoção
 - Remover D



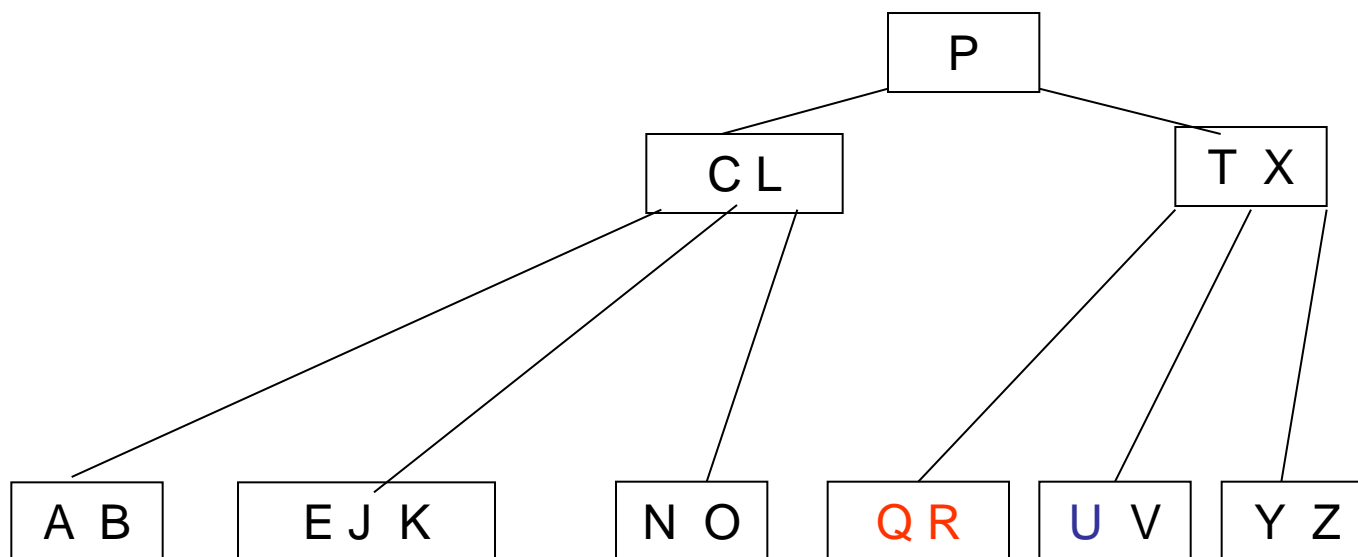
Árvore B

- Exemplo remoção
 - Remover S



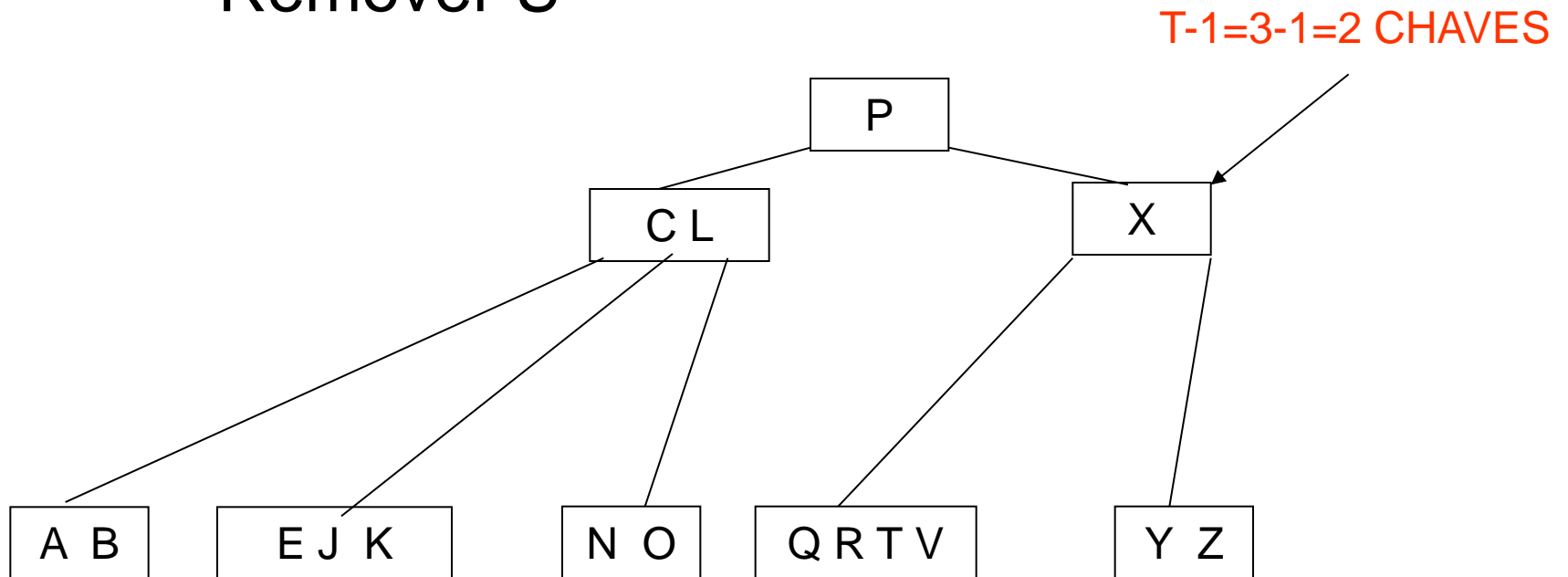
Árvore B

- Exemplo remoção
 - Remover U



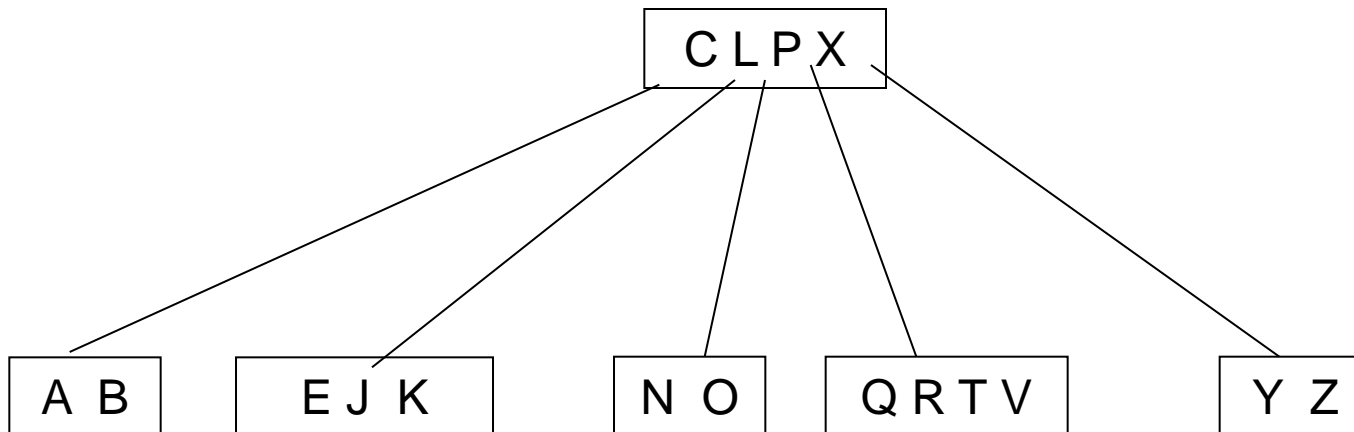
Árvore B

- Exemplo remoção
 - Remover U



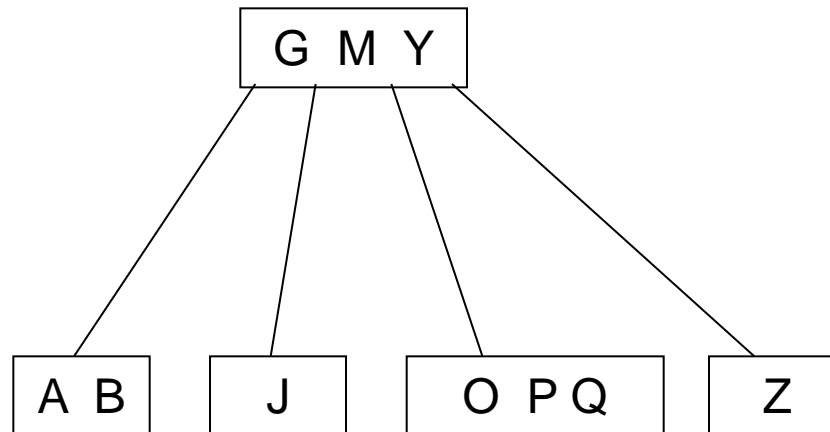
Árvore B

- Exemplo remoção
 - Remover



Árvore B

- Árvore 2-3-4
 - Ocorre quando $t=2$ e é a menor árvore B possível

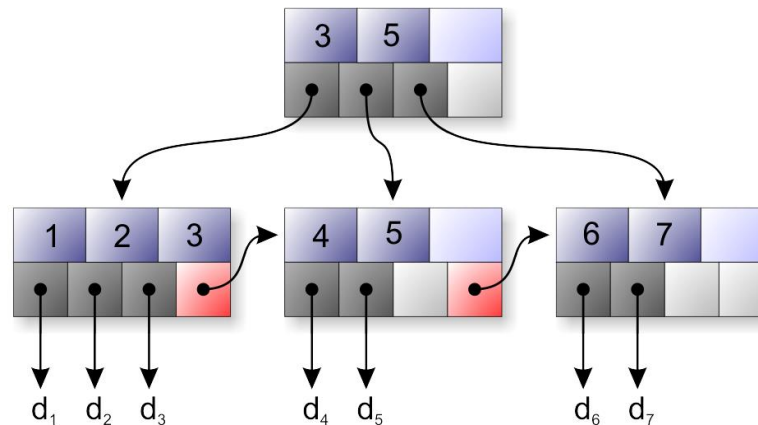


- Mín: $t-1=1$ chave e 2 filhos
- Máx: $2t-1=3$ chaves e 4 filhos

Árvore B

- **Árvore B+**

- São árvores B que contêm dados somente nos nós folhas e existe uma ligação entre os irmãos folhas adjacentes para facilitar a busca



- São bastante utilizados para criar índices de banco de dados
- O NTFS do Windows usa árvores B+