Exclusão em Árvores B

Paulo Ricardo Lisboa de Almeida





O algoritmo de exclusão é um pouco mais complicado que o de inserção.



Incluir pode causar um overflow no tamanho máximo do nodo.

Precisamos fazer um split para criar mais espaço.

Uma exclusão pode causar um underflow no tamanho mínimo.

Precisamos transplantar chaves ou, no pior caso, fundir (merge) nodos.

O algoritmo é uma modificação do buscar Arvore B.

Buscamos a chave a ser excluída.

A cada nível que descemos, precisamos garantir que o nodo tenha pelo menos t chaves.

Atenção: a carga/descarga dos nodos será ocultada nos algoritmos.

```
função excluirArvoreB(T,x,k)
entrada: árvore B T, nodo x de onde começar a busca, e
chave a ser excluída k.
saída: a chave k é excluída da árvore B, caso exista.
i = 1
enquanto i \le x.n e k > x.chave[i]
     i = i+1
se i \le x.n e k == x.chave[i] //chave encontrada
     se x é folha
           remova k de x
           retorne
     senão
           //...
senão
     se x é folha
           retorne //chave não encontrada
retorne excluirArvoreB(x.filhos[i],k)
```

Caso 1, a chave está em uma folha, ou não está na árvore.

função excluirArvoreB(T,k)

```
i = 1
enquanto i ≤ x.n e k > x.chave[i]
      i = i+1
se i \le x.n e k == x.chave[i]
      se x é folha
            remova k de x
            retorne
      senão
            //...
senão
      se x é folha
            retorne
```

retorne excluirArvoreB(x.filhos[i],k)

Raiz n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =

n = 2 chaves = A B

n = 3chaves = D E F

n = 3chaves = | n = 2 chaves = N 0 n = 3 chaves = Q R S n = 2 chaves = U V

n = 2 chaves = Y

```
função excluirArvoreB(T,k)
i = 1
enquanto i \le x.n e k > x.chave[i]
      i = i+1
se i ≤ x.n e k == x.chave[i]
      se x é folha
            remova k de x
            retorne
      senão
            //...
senão
      se x é folha
            retorne
retorne excluirArvoreB(x.filhos[i],k)
```

n = 2

chaves = A B

n = 3

chaves = D E F

n = 3

chaves = J

```
Raiz
                 n = 1
                 chaves =
                  filhos =
                              n = 2
n = 3
chaves =
                              chaves =
filhos =
                               filhos =
```

n = 2

n = 3 chaves = N 0 chaves = Q R S

n = 2 chaves = U V

n = 2 chaves = Y

```
função excluirArvoreB(T,k)

i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]
    se x é folha
        remova k de x
    retorne
    senão
        //...
senão
    se x é folha
        retorne
retorne excluirArvoreB(x.filhos[i],k)
```

Raiz n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =

n = 2 n = 3 chaves = A B chave

n = 3 1 2 3 chaves = J K L n = 2 $\frac{1}{\text{chaves}} = \frac{2}{N} = 0$

 n = 2

chaves = UVV

n = 2chaves = $\begin{bmatrix} 1 & 2 \\ Y & Z \end{bmatrix}$

função excluirArvoreB(T,k)

```
i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]
    se x é folha
        remova k de x
        retorne
    senão
        //...
senão
    se x é folha
        retorne
retorne excluirArvoreB(x.filhos[i],k)
```

Raiz · n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =

n = 2

 n = 2chaves = $\begin{bmatrix} 1 & 2 \\ U & V \end{bmatrix}$

n = 2chaves = $\begin{bmatrix} 1 & 2 \\ Y & Z \end{bmatrix}$

```
função excluirArvoreB(T,k)

i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]

    se x é folha
        remova k de x
        retorne
    senão
        //...
senão
    se x é folha
        retorne
retorne excluirArvoreB(x.filhos[i],k)
```

Raiz n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =

n = 2

n = 3 $1 \quad 2 \quad 3$ $chaves = \boxed{Q} \quad R \quad S$

n = 2chaves = $\begin{bmatrix} 1 & 2 \\ V & V \end{bmatrix}$

n = Zchaves = $\begin{bmatrix} 1 & 2 \\ Y & Z \end{bmatrix}$

```
função excluirArvoreB(T,k)
i = 1
enquanto i \le x.n e k > x.chave[i]
      i = i+1
se i \le x.n e k == x.chave[i]
      se x é folha
            remova k de x
            retorne
      senão
            //...
senão
      se x é folha
            retorne
retorne excluirArvoreB(x.filhos[i],k)
```

n = 2

chaves = A B

n = 3

chaves = D E F

n = 3

chaves = |

Raiz · n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =

n = 2

chaves = N 0

n = 3

n = 2 chaves = Q R S chaves = U V

n = 2 chaves = Y

função excluirArvoreB(T,k)

```
i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]
    se x é folha
        remova k de x
        retorne
    senão
        //...
senão
    se x é folha
    retorne
```

retorne excluirArvoreB(x.filhos[i],k)

Raiz n = 1 chaves = filhos = n = 2 n = 3chaves = chaves = filhos = filhos =


```
função excluirArvoreB(T,k)
i = 1
enquanto i \le x.n e k > x.chave[i]
      i = i+1
se i ≤ x.n e k == x.chave[i]
      se x é folha
            remova k de x
            retorne
      senão
            //...
senão
      se x é folha
            retorne
retorne excluirArvoreB(x.filhos[i],k)
```

n = 2

```
Raiz
                 n = 1
                 chaves =
                 filhos =
                              n = 2
n = 3
chaves =
                              chaves =
filhos =
                              filhos =
```

n = 2 n = 3 n = 3n = 3chaves = N 0 chaves = A B chaves = D E F chaves = | J

n = 2 n = 2 chaves = U V chaves = Q R S chaves = Y

```
função excluirArvoreB(T,k)

i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]
    se x é folha
        remova k de x
        retorne
    senão
        //...
senão
    se x é folha
    retorne
```

retorne excluirArvoreB(x.filhos[i],k)

```
Raiz
                 n = 1
                 chaves =
                 filhos =
                              n = 2
n = 3
chaves =
                              chaves =
filhos =
                              filhos =
```

n = 2 $chaves = \begin{bmatrix} 1 & 2 \\ A & B \end{bmatrix}$ chave

 $\text{chaves} = \begin{array}{c|c} 1 & 2 & 3 \\ \hline D & E & F \\ \hline \end{array}$

 n = 3 $1 \quad 2 \quad 3$ $chaves = \boxed{Q} \quad R \quad S$

n = 2

chaves = UVV

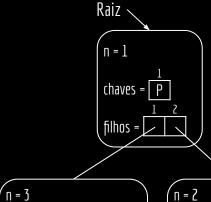
n = 2chaves = $\begin{bmatrix} 1 & 2 \\ Y & Z \end{bmatrix}$

```
função excluirArvoreB(T,k)
i = 1
enquanto i \le x.n e k > x.chave[i]
      i = i+1
se i \le x.n e k == x.chave[i]
      se x é folha
```

remova k de x retorne senão //...

senão se x é folha retorne

retorne excluirArvoreB(x.filhos[i],k)



chaves = filhos = chaves = filhos =

n = 2

n = 2 chaves = U V chaves = Y

```
n = 2
chaves = A B
```

n = 2 chaves = D E n = 3chaves = | |

chaves = N 0

n = 3

chaves = Q R S

n = 2

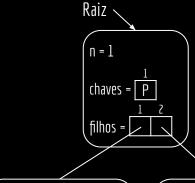
```
função excluirArvoreB(T,k)
i = 1
enquanto i ≤ x.n e k > x.chave[i]
    i = i+1
se i ≤ x.n e k == x.chave[i]
    se x é folha
        remova k de x
```

retorne senão //...

senão

se x é folha retorne

 $retorne \ \textbf{excluirArvoreB}(\texttt{x.filhos[i],k})$



 $\begin{array}{c}
\text{n = 3} \\
\text{chaves = } \begin{array}{c|c}
\hline
\text{C} & G & M \\
\hline
\end{array}$ $\begin{array}{c|c}
\hline
\text{filhos = } \\
\end{array}$

n = 2

chaves = UVV

n = 2chaves = $\begin{bmatrix} 1 & 2 \\ Y & Z \end{bmatrix}$

n = 2chaves = $\begin{bmatrix} 1 & 2 \\ A & B \end{bmatrix}$

 $\begin{array}{ccc}
1 & 2 \\
\text{chaves} & = & D & E
\end{array}$

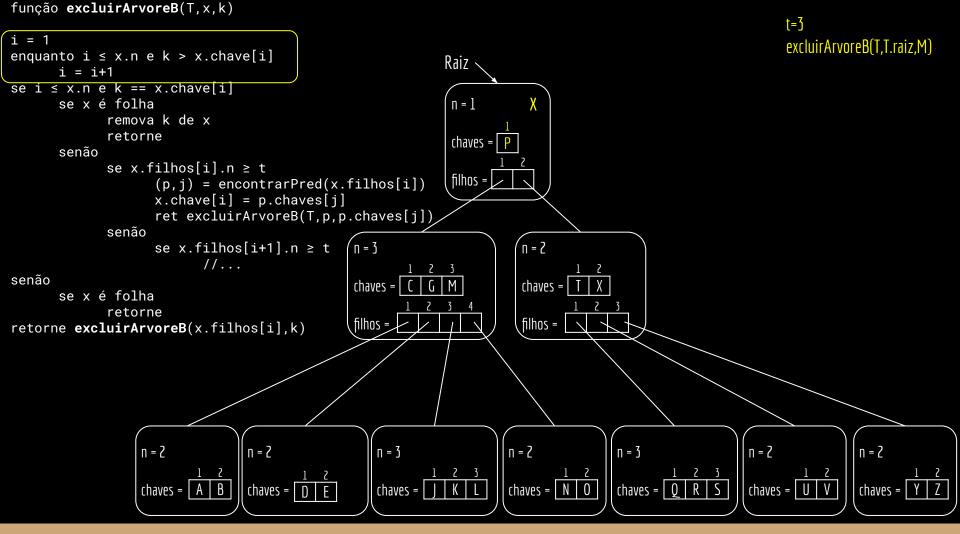
n = 2

n = 2

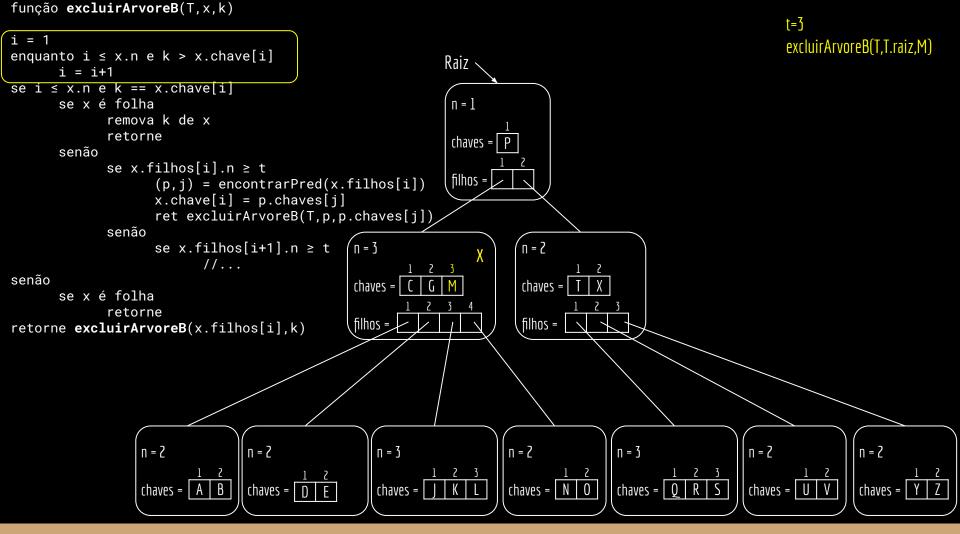
n = 3

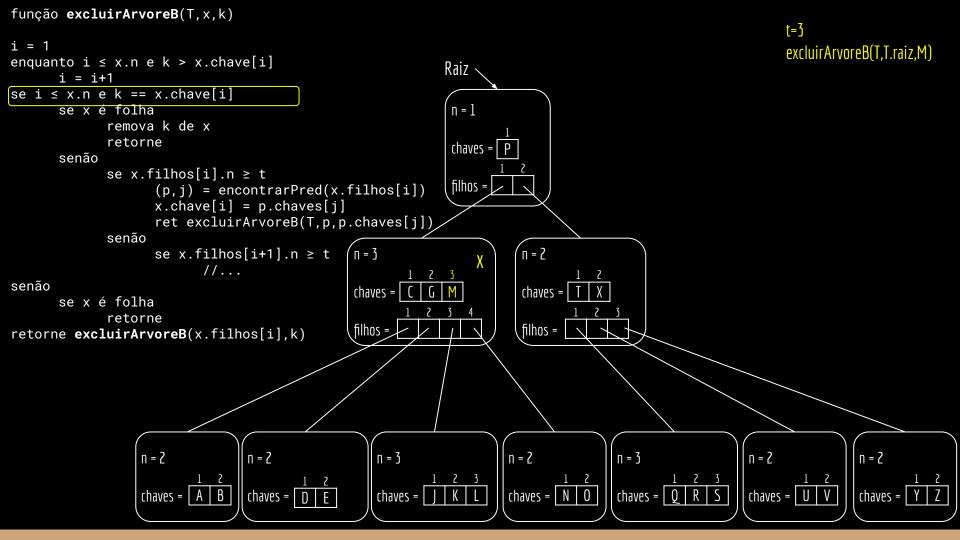
```
função excluirArvoreB(T,x,k)
i = 1
enquanto i \le x.n \in k > x.\underline{chave[i]}
      i = i+1
se i \le x.n e k == x.chave[i] //chave encontrada
      se x é folha
            remova k de x
            retorne
      senão
            se x.filhos[i].n ≥ t
                   (p,j) = encontrarPred(x.filhos[i]) //Predecessor de k
                  x.chave[i] = p.chaves[j]
                  retorne excluirArvoreB(T,p,p.chaves[j]) // recursão
            senão
                  se x.filhos[i+1].n \geq t
                         //...
senão
      se x é folha
                                                                      Caso 2, a chave está em um nodo interno.
            retorne //chave não encontrada
                                                                      2a. Subárvore esquerda possui pelo menos t chaves.
retorne excluirArvoreB(x.filhos[i],k)
                                                                      2b. Subárvore esquerda possui t-1 chaves. Subárvore direita
```

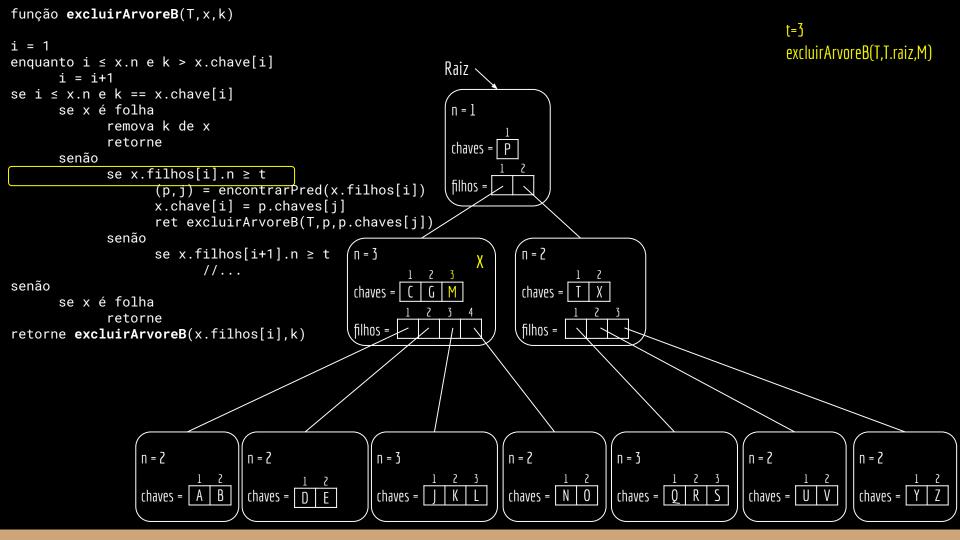
possui pelo menos t chaves. Simétrico a 2a.

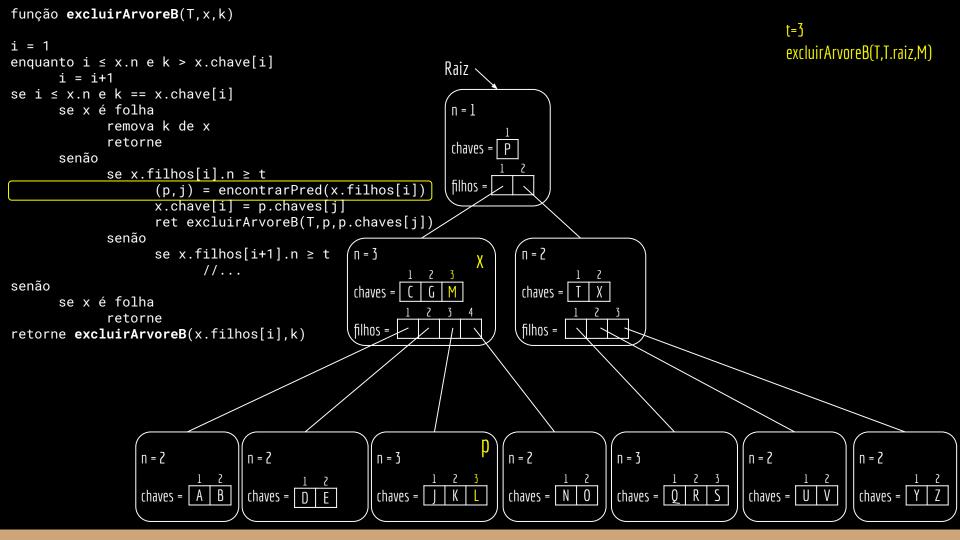


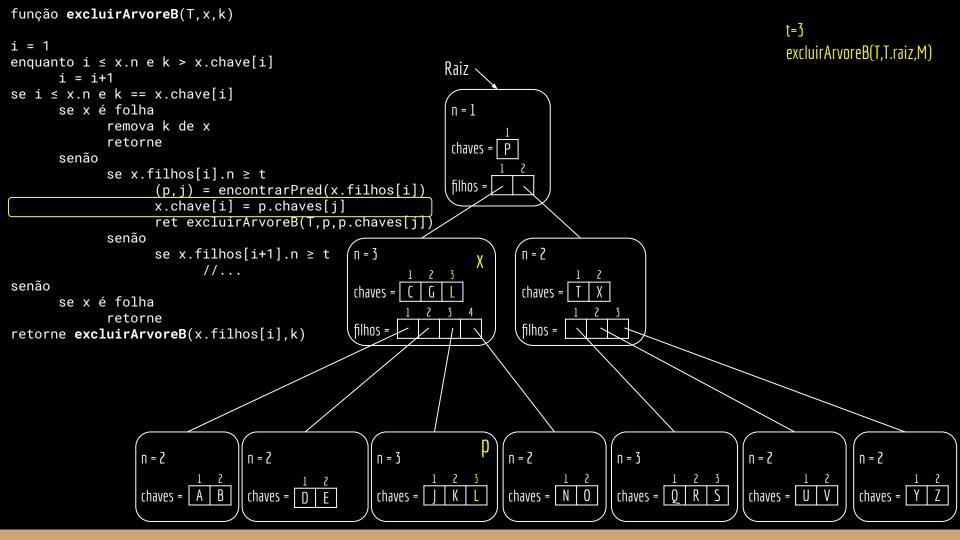
```
t=3
i = 1
                                                                                                                excluirArvoreB(T,T.raiz,M)
enquanto i \le x.n \in k > x.chave[i]
                                                               Raiz ·
       i = i+1
se i \le x.n e k == x.chave[i]
                                                               n = 1
       se x é folha
              remova k de x
              retorne
                                                               chaves =
       senão
              se x.filhos[i].n ≥ t
                                                               filhos =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     ret excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                         n = 2
                     se x.filhos[i+1].n \ge t
                            //...
senão
                                                 chaves =
                                                                          chaves =
       se x é folha
              retorne
                                                                          filhos =
                                                  filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                       n = 2
                                                                                       n = 3
                                                                                                          n = 2
                                                                                                                          n = 2
                                  n = 2
                                                    n = 3
                                                                                                          chaves = U V
                  chaves = A
                                                                        chaves = N 0
                                                                                       chaves = Q R S
                                  chaves = D E
                                                    chaves = |
                                                                                                                           chaves = | Y
```











```
t=3
i = 1
                                                                                                                excluirArvoreB(T,T.raiz,M)
enquanto i \le x.n \in k > x.chave[i]
                                                               Raiz ·
      i = i+1
se i \le x.n e k == x.chave[i]
                                                               n = 1
       se x é folha
              remova k de x
              retorne
                                                               chaves =
       senão
              se x.filhos[i].n ≥ t
                                                                filhos =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                    ret excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                          n = 2
                     se x.filhos[i+1].n \ge t
                                                 n = 3
                            //...
senão
                                                 chaves =
                                                                          chaves =
      se x é folha
              retorne
                                                                          filhos =
                                                  filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                       n = 2
                                                                                       n = 3
                                                                                                           n = 2
                                                                                                                           n = 2
                                  n = 2
                                                     n = 3
                                                                        chaves = N 0
                                                                                                           chaves = U V
                  chaves = A
                                                                                        chaves = Q R S
                                  chaves = D E
                                                     chaves =
                                                                                                                           chaves = | Y
```

```
t=3
i = 1
                                                                                                                excluirArvoreB(...,L)
enquanto i \le x.n \in k > x.chave[i]
                                                               Raiz ·
       i = i+1
se i \le x.n e k == x.chave[i]
                                                                n = 1
       se x é folha
              remova k de x
              retorne
                                                                chaves =
       senão
              se x.filhos[i].n ≥ t
                                                                filhos =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     ret excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                          n = 2
                     se x.filhos[i+1].n \ge t
                            //...
senão
                                                 chaves =
                                                                          chaves =
       se x é folha
              retorne
                                                                          filhos =
                                                  filhos =
retorne excluirArvoreB(x.filhos[i],k)
                   n = 2
                                                                        n = 2
                                                                                        n = 3
                                                                                                           n = 2
                                                                                                                           n = 2
                                  n = 2
                                                     n = 3
                                                                        chaves = N 0
                                                                                                           chaves = U V
                   chaves = A
                                                                                        chaves = Q R S
                                  chaves = D E
                                                     chaves =
                                                                                                                           chaves = | Y
```

```
t=3
i = 1
                                                                                                               excluirArvoreB(...,L)
enquanto i \le x.n \in k > x.chave[i]
                                                              Raiz ·
       i = i+1
se i ≤ x.n e k == x.chave[i]
                                                               n = 1
      se x é folha
             remova k de x
             retorne
                                                               chaves =
       senão
             se x.filhos[i].n ≥ t
                                                               filhos =
                    (p,j) = encontrarPred(x.filhos[i])
                    x.chave[i] = p.chaves[j]
                    ret excluirArvoreB(T,p,p.chaves[j])
             senão
                                                                         n = 2
                    se x.filhos[i+1].n \ge t
                           //...
senão
                                                 chaves =
                                                                         chaves =
      se x é folha
             retorne
                                                                         filhos =
                                                 filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                       n = 2
                                                                                       n = 3
                                                                                                          n = 2
                                                                                                                          n = 2
                                  n = 2
                                                    n = 3
                                                                                                          chaves = U V
                  chaves = A
                                                                       chaves = N O
                                                                                       chaves = Q R S
                                  chaves = D E
                                                    chaves =
                                                                                                                          chaves = | Y
```

```
t=3
i = 1
                                                                                                               excluirArvoreB(...,L)
enquanto i \le x.n \in k > x.chave[i]
                                                              Raiz
      i = i+1
se i \le x.n e k == x.chave[i]
                                                               n = 1
      se x é folha
              remova k de x
              retorne
                                                               chaves =
       senão
             se x.filhos[i].n ≥ t
                                                               filhos =
                    (p,j) = encontrarPred(x.filhos[i])
                    x.chave[i] = p.chaves[j]
                    ret excluirArvoreB(T,p,p.chaves[j])
             senão
                                                                         n = 2
                    se x.filhos[i+1].n \ge t
                           //...
senão
                                                 chaves =
                                                                         chaves =
       se x é folha
              retorne
                                                                          filhos =
                                                 filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                       n = 2
                                                                                       n = 3
                                                                                                          n = 2
                                                                                                                          n = 2
                                  n = 2
                                                    n = 3
                                                                                                          chaves = U V
                  chaves = A
                                                                       chaves = N O
                                                                                       chaves = Q R S
                                  chaves = D E
                                                    chaves =
                                                                                                                          chaves = | Y
```

```
função excluirArvoreB(T,x,k)
                                                                                                              t=3
i = 1
                                                                                                              excluirArvoreB(...,L)
enquanto i \le x.n \in k > x.chave[i]
                                                             Raiz
      i = i+1
se i \le x.n e k == x.chave[i]
                                                              n = 1
      se x é folha
             remova k de x
             retorne
                                                              chaves =
       senão
             se x.filhos[i].n ≥ t
                                                               filhos =
                    (p,j) = encontrarPred(x.filhos[i])
                    x.chave[i] = p.chaves[j]
                    ret excluirArvoreB(T,p,p.chaves[j])
             senão
                                                                        n = 2
                    se x.filhos[i+1].n \ge t
                           //...
senão
                                                chaves =
                                                                         chaves =
      se x é folha
             retorne
                                                                         filhos =
                                                 filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                      n = 2
                                                                                      n = 3
                                                                                                         n = 2
                                                                                                                         n = 2
                                 n = 2
                                                    n = 3
                                                                      chaves = N 0
                                                                                                         chaves = U V
                  chaves = A
                                                                                      chaves = Q R S
                                 chaves = D E
                                                    chaves = | | K
                                                                                                                         chaves = | Y
```

```
função excluirArvoreB(T,x,k)
                                                                                                              t=3
i = 1
                                                                                                              excluirArvoreB(...,L)
enquanto i \le x.n \in k > x.chave[i]
                                                              Raiz ·
      i = i+1
se i \le x.n e k == x.chave[i]
                                                              n = 1
       se x é folha
             remova k de x
             retorne
                                                               chaves =
       senão
             se x.filhos[i].n ≥ t
                                                               filhos =
                    (p,j) = encontrarPred(x.filhos[i])
                    x.chave[i] = p.chaves[j]
                    ret excluirArvoreB(T,p,p.chaves[j])
             senão
                                                                         n = 2
                    se x.filhos[i+1].n \ge t
                           //...
senão
                                                chaves =
                                                                         chaves =
      se x é folha
             retorne
                                                                         filhos =
                                                 filhos =
retorne excluirArvoreB(x.filhos[i],k)
                  n = 2
                                                                      n = 2
                                                                                      n = 3
                                                                                                         n = 2
                                                                                                                         n = 2
                                 n = 2
                                                    n = 3
                                                                                                         chaves = U V
                  chaves = A
                                                                       chaves = N O
                                                                                      chaves = Q R S
                                 chaves = D E
                                                    chaves = | K
                                                                                                                         chaves = | Y
```

```
enquanto i \le x.n e k > x.chave[i]
                                                               Caso 2c: ambas subárvores têm t-1 chaves. Passar os dados
     i = i+1
se i \le x.n e k == x.chave[i] //chave encontrada
                                                                (merge) de x.filhos[i+1] para x.filhos[i], juntamente com a chave k.
     se x é folha
                                                               Chamar a exclusão recursivamente.
           remova k de x
           retorne
     senão
           se x.filhos[i].n ≥ t
                  (p,j) = encontrarPred(x.filhos[i]) //Predecessor de k
                 x.chave[i] = p.chaves[j]
                 retorne excluirArvoreB(T,p,p.chaves[j]) // recursão
           senão
                 se x.filhos[i+1].n \geq t
                        //...
                 senão
                       remova k de x
                        //x.filhos[i] vai receber tudo de x.filhos[i+1] e k
                       merge (x.filhos[i], x.filhos[i+1], k)
                       excluir nodo x.filhos[i+1] da memória
                       excluirArvoreB(x.filhos[i],k)
                       se x.n < 1 //x era raiz e pode não ter mais chaves
                             T.raiz = x.filhos[1]
                             exclua o nodo x da memória
                       retorne
senão
     se x é folha
           retorne //chave não encontrada
retorne excluirArvoreB(x.filhos[i],k)
```

i = 1

```
funcão excluirArvoreB(T.x.k)
i = 1
                                                                                                                     t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i ≤ x.n e k == x.c<u>have[i</u>]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = [
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                           n = 2
                                           n = 2
                                                                           n = 2
                                                                                           n = 3
                                                           n = 3
                           chaves = A
                                                                                           chaves = Q R S
                                                                                                               chaves = U
                                            chaves = D
                                                           chaves =
                                                                           chaves = N O
                                                                                                                                chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos =
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                           n = 2
                                           n = 2
                                                           n = 3
                                                                          n = 2
                                                                                           n = 3
                                                                                                               chaves = U V
                           chaves = A
                                                                                           chaves = Q R S
                                            chaves = D
                                                           chaves = J
                                                                           chaves = N O
                                                                                                                                chaves = | Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i ≤ x.n e k == x.c<u>have[i</u>]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                      x.chave[i] = p.chaves[j]
                      retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = [
              senão
                      se x.filhos[i+1].n \ge t
                             //...
                      senão
                                                                                                        n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                               filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                           n = 2
                                            n = 2
                                                                           n = 2
                                                                                           n = 3
                                                           n = 3
                           chaves = A
                                                                                            chaves = Q R S
                                                                                                                chaves = U
                                            chaves = D
                                                           chaves = ]
                                                                           chaves = N O
                                                                                                                                chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \ e \ k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i ≤ x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                      x.chave[i] = p.chaves[j]
                      retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos =
              senão
                      se x.filhos[i+1].n \ge t
                             //...
                      senão
                                                                                                        n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves :
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                               filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                 n = 2
                           n = 2
                                            n = 2
                                                           n = 3
                                                                           n = 2
                                                                                           n = 3
                                                                                                                chaves = U
                           chaves = A
                                                                           chaves = N O
                                                                                            chaves = Q R S
                                            chaves = D
                                                           chaves = ]
                                                                                                                                 chaves = | Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = [
              senão
                     se x.filhos[i+1].n ≥ t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                           n = 2
                                           n = 2
                                                           n = 3
                                                                          n = 2
                                                                                           n = 3
                                                                                                               chaves = U
                           chaves = A
                                                                                           chaves = Q R S
                                           chaves = D
                                                           chaves = ]
                                                                           chaves = N O
                                                                                                                                chaves = | Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                      x.chave[i] = p.chaves[j]
                      retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = |
              senão
                      se x.filhos[i+1].n \ge t
                             //...
                      senão
                                                                                                        n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves :
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                               filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                           n = 2
                                                                                                                n = 2
                                                                                                                                n = 2
                           n = 2
                                                                                           n = 3
                                            n = 5
                                                                    n = 3
                           chaves = A
                                                                           chaves = N O
                                                                                            chaves = Q R S
                                                                                                                chaves = U
                                                                                                                                 chaves = Y
                                            chaves = D
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = [
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i]. x.filhos[i+1]. k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                                n = 2
                                                                           n = 2
                                                                                           n = 3
                                                 n = 5
                                                                                                               chaves = U
                                                                           chaves = N O
                                                                                           chaves = Q R S
                                chaves = A
                                                 chaves = D
                                                                                                                                chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(T,T.raiz,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos =
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                               n = 2
                                                                                                                                n = 2
                                n = 2
                                                                           n = 2
                                                                                           n = 3
                                                n = 5
                                                                                                               chaves = U
                                                                           chaves = N O
                                                                                           chaves = Q R S
                                chaves = A
                                                chaves = D
                                                                                                                                chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                    t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                    excluirArvoreB(...,G)
       i = i+1
                                                                                           Raiz
se i ≤ x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                            n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                            chaves =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                            filhos = [
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                             n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                             chaves =
                                                                                                       chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                       filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                          n = 2
                                                                                                              n = 2
                                                                                                                               n = 2
                                n = 2
                                                n = 5
                                                                                          n = 3
                                                                          chaves = N 0
                                                                                           chaves = Q R S
                                                                                                              chaves = U
                                chaves = A
                                                                                                                               chaves = Y
                                                chaves =
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                    t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                    excluirArvoreB(...,G)
       i = i+1
                                                                                            Raiz
se i ≤ x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                            n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos = [
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                             n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                       chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                       filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                          n = 2
                                                                                                               n = 2
                                                                                                                               n = 2
                                n = 2
                                                n = 5
                                                                                           n = 3
                                                                                                               chaves = U
                                                                          chaves = N O
                                                                                           chaves = Q R S
                                chaves = A
                                                                                                                               chaves = Y
                                                chaves =
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                     t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                     excluirArvoreB(...,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                             n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                             chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                             filhos =
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                              n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                             excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                        chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                        filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                          n = 2
                                                                                                               n = 2
                                                                                                                               n = 2
                                n = 2
                                                n = 5
                                                                                           n = 3
                                                                           chaves = N O
                                                                                           chaves = Q R S
                                                                                                               chaves = U
                                chaves = A
                                                                                                                                chaves = Y
                                                 chaves =
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                    t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                    excluirArvoreB(...,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                            n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                            chaves =
                      (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                            filhos =
              senão
                     se x.filhos[i+1].n \ge t
                             //...
                     senão
                                                                                                       n = 2
                                                                             n = 3
                             remova k de x
                             merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                       chaves
                             excluirArvoreB(x.filhos[i],k)
                             se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                       filhos =
                                    exclua o nodo x da memória
                             retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                          n = 2
                                                                                                               n = 2
                                                                                                                               n = 2
                                     n = 2
                                                                                           n = 3
                                                     n = 4
                                                                          chaves = N 0
                                     chaves = A B
                                                                                           chaves = Q R S
                                                                                                               chaves = U
                                                     chaves = D
                                                                                                                               chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                    t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                    excluirArvoreB(...,G)
       i = i+1
                                                                                            Raiz
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
                                                                                            n = 1
              retorne
       senão
              se x.filhos[i].n ≥ t
                                                                                            chaves =
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
                                                                                            filhos =
              senão
                     se x.filhos[i+1].n \ge t
                            //...
                     senão
                                                                                                       n = 2
                                                                             n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                              chaves =
                                                                                                       chaves
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                    T.raiz = x.filhos[1]
                                                                              filhos =
                                                                                                       filhos =
                                    exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                          n = 2
                                                                                                               n = 2
                                                                                                                               n = 2
                                     n = 2
                                                                                           n = 3
                                                     n = 4
                                                                          chaves = N 0
                                     chaves = A B
                                                                                                               chaves = U
                                                                                           chaves = Q R S
                                                                                                                               chaves = Y
                                                     chaves = D
```

```
i = i+1
se i ≤ x.n e k == x.chave[i] //chave encontrada
     //...
senão
     se x é folha
           retorne //chave não encontrada
     senão
           se x.filhos[i].n < t</pre>
                 b = irmaoImediatoComMaisChaves(x,i)
                 se b.n ≥ t
                      passar x.chaves[i] para x.filhos[i]
                      se b era irmão direito
                            passar b.chaves[1] para x
                      senão
                            passar b.chaves[b.n] para x
retorne excluirArvoreB(T,x.filhos[i],k)
```

enquanto $i \le x.n e k > x.chave[i]$

continuar a busca, garantir que todo nodo visitado possui pelo menos t chaves.

Caso 3a. O próximo nodo x.filhos[i] possui t-1 chaves, mas tem um irmão imediato com pelo menos t chaves. Mova uma chave de x para x.filhos[i], e transfira uma chave de um dos irmãos para x.

Caso 3: Chegamos em um nodo interno que não possui k. Antes de

```
função excluirArvoreB(T,x,k)
                                                                  Caso 3: Chegamos em um nodo interno que não possui k. Antes de
enquanto i \le x.n \in k > x.chave[i]
      i = i+1
                                                                  continuar a busca, garantir que todo nodo visitado possui pelo
se i \le x.n e k == x.chave[i] //chave encontrada
                                                                  menos t chaves.
      //...
senão
      se x é folha
            retorne //chave não encontrada
      senão
           se x.filhos[i].n < t</pre>
                  b = irmaoImediatoComMaisChaves(x,i)
                  se b.n ≥ t
                        passar x.chaves[i] para x.filhos[i]
                                                                  nodo.
                        se b era irmão direito
                              passar b.chaves[1] para x
                        senão
                              passar b.chaves[b.n] para x
                  senão
                        merge (x.filhos[i], b, x.chaves[i])
                        excluir nodo b da memória
                        remova x.chaves[i] de x
                        se x.n < 1 //x era raiz e pode não ter mais chaves
                              T.raiz = x.filhos[1]
                              exclua o nodo x da memória
                              retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
```

Caso 3a. O próximo nodo x.filhos[i] possui t-1 chaves, mas tem um

irmão imediato com pelo menos t chaves. Mova uma chave de x para x.filhos[i], e transfira uma chave de um dos irmãos para x. Caso 3b. Nenhum irmão imediato possui pelo menos t chaves. Passar os dados (merge) de um irmão imediato para x.filhos[i], juntamente com a chave x.chaves[i] que será o mediano do novo

Caso3b-1: x era a raiz e ficou sem nodos.

```
Caso 3: Chegamos em um nodo interno que não possui k. Antes de
enquanto i \le x.n \in k > x.chave[i]
      i = i+1
se i \le x.n e k == x.chave[i] //chave encontrada
                                                                  menos t chaves.
      //...
senão
      se x é folha
            retorne //chave não encontrada
      senão
           se x.filhos[i].n < t</pre>
                  b = irmaoImediatoComMaisChaves(x,i)
                  se b.n ≥ t
                        passar x.chaves[i] para x.filhos[i]
                                                                  nodo.
                        se b era irmão direito
                                                                        Caso3b-1: x era a raiz e ficou sem nodos.
                              passar b.chaves[1] para x
                        senão
                              passar b.chaves[b.n] para x
                  senão
                       merge (x.filhos[i], b, x.chaves[i])
                        excluir nodo b da memória
                        remova x.chaves[i] de x
                        se x.n < 1 //x era raiz e pode não ter mais chaves
                              T.raiz = x.filhos[1]
                              exclua o nodo x da memória
                              retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T, x.filhos[i],k)
```

continuar a busca, garantir que todo nodo visitado possui pelo Caso 3a. O próximo nodo x.filhos[i] possui t-1 chaves, mas tem um irmão imediato com pelo menos t chaves. Mova uma chave de x para x.filhos[i], e transfira uma chave de um dos irmãos para x. Caso 3b. Nenhum irmão imediato possui pelo menos t chaves. Passar os dados (merge) de um irmão imediato para x.filhos[i], juntamente com a chave x.chaves[i] que será o mediano do novo

Caso 3c. O nodo x.filhos[i] possui t ou mais chaves.

```
t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                     Raiz
se i \le x.n e k == x.chave[i]
       //...
senão
                                                                                     n = 1
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                    b = irmaoImediatoComMaisChaves(x,i)
                                                                                      filhos =
                    se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                        n = 3
                                                                                               n = 2
                          senão
                                 passar b.chaves[b.n] para x
                                                                        chaves =
                                                                                               chaves =
                    senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                        filhos =
                                                                                               filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne excluirArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                    n = 3
                                                                     n = 2
                                                                                                      n = 2
                                                                                                                      n = 2
                                  n = 2
                                                 n = 4
                                                                     chaves = N O
                                                                                                      chaves = U V
                                                                                    chaves = Q R S
                                  chaves = A B
                                                 chaves = D E
                                                                                                                      chaves = Y
```

i = 1

```
i = 1
                                                                                                           t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                    Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 1
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos =
                   se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                                                                                               n = 2
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                                                                                               chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                        filhos =
                                                                                               filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne excluirArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                   n = 3
                                                                    n = 2
                                                                                                      n = 2
                                                                                                                     n = 2
                                  n = 2
                                                 n = 4
                                                                    chaves = N O
                                                                                                      chaves = U V
                                  chaves = A B
                                                                                   chaves = Q R S
                                                 chaves = D E
                                                                                                                     chaves = Y
```

```
i = 1
                                                                                                            t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                            excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                     Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 1
      se x é folha
             retorne //chave não encontrada
                                                                                      chaves =
      senão
             se x.filhos[i].n < t</pre>
                    b = irmaoImediatoComMaisChaves(x,i)
                                                                                      filhos =
                    se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                        n = 3
                                                                                               n = 2
                          senão
                                 passar b.chaves[b.n] para x
                                                                        chaves =
                                                                                                chaves =
                    senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                        filhos =
                                                                                                filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne excluirArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                    n = 3
                                                                     n = 2
                                                                                                      n = 2
                                                                                                                      n = 2
                                  n = 2
                                                 n = 4
                                                                     chaves = N O
                                                                                                       chaves = U V
                                                                                    chaves = Q R S
                                  chaves = A B
                                                 chaves = D E
                                                                                                                      chaves = Y
```

```
t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                    Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 1
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos = /
                   se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne exclujrArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                                                     n = 2
                                                                    n = 2
                                                                                   n = 3
                                                                                                     n = 2
                                  n = 2
                                                n = 4
                                                                    chaves = N O
                                                                                                     chaves = U V
                                  chaves = A B
                                                                                   chaves = Q R S
                                                 chaves = D E
                                                                                                                     chaves = Y
```

i = 1

```
i = 1
                                                                                                           t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                    Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 1
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos = 🖊
                   se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne exclujrArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                   n = 3
                                                                                                                     n = 2
                                                                    n = 2
                                                                                                     n = 2
                                  n = 2
                                                n = 4
                                                                    chaves = N O
                                                                                                     chaves = U V
                                  chaves = A B
                                                                                   chaves = Q R S
                                                 chaves = D E
                                                                                                                     chaves = Y
```

```
i = 1
                                                                                                           t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                    Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 0
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos = /
                   se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne exclujrArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                                                     n = 2
                                                                    n = 2
                                                                                   n = 3
                                                                                                     n = 2
                                  n = 2
                                                n = 4
                                                                    chaves = N O
                                                                                   chaves = Q R S
                                                                                                     chaves = U V
                                  chaves = A B
                                                 chaves = D E
                                                                                                                     chaves = Y
```

```
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T,T.raiz,D)
      i = i+1
                                                                                    Raiz
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                     n = 0
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos = /
                   se b.n ≥ t
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1
                                 exclua o nodo x da memória
                                 retorne exclujrArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                   n = 3
                                                                                                                     n = 2
                                                                    n = 2
                                                                                                     n = 2
                                  n = 2
                                                n = 4
                                                                    chaves = N O
                                                                                   chaves = Q R S
                                                                                                     chaves = U V
                                  chaves = A B
                                                 chaves = D E
                                                                                                                     chaves = Y
```

```
i = 1
                                                                                                           t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                           excluirArvoreB(T,T.raiz,D)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
                                                                                    n = 0
      se x é folha
             retorne //chave não encontrada
                                                                                     chaves =
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                                                                                     filhos = /
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memoria
                                 retorne exclujrArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                                                     n = 2
                                                                    n = 2
                                                                                   n = 3
                                                                                                     n = 2
                                  n = 2
                                                n = 4
                                                                                   chaves = Q R S
                                                                                                     chaves = U V
                                  chaves = A B
                                                                    chaves = N O
                                                 chaves = D E
                                                                                                                     chaves = Y
```

```
i = 1
                                                                                                         t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                         excluirArvoreB(T,T.raiz,D)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                      chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                T.raiz = x.filhos[1]
                                exclua o nodo x da memória
                                 retorne exclujrarvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                                                   n = 2
                                                                   n = 2
                                                                                  n = 3
                                                                                                    n = 2
                                  n = 2
                                                n = 4
                                                                                  chaves = Q R S
                                                                                                    chaves = U V
                                  chaves = A B
                                                                   chaves = N O
                                                chaves = D E
                                                                                                                    chaves = Y
```

```
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T,T.raiz,D)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                      chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                T.raiz = x.filhos[1]
                                exclua o nodo x da memória
                                retorne excluirArvoreB(T,T,raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                                                    n = 2
                                                                   n = 2
                                                                                  n = 3
                                                                                                    n = 2
                                  n = 2
                                                n = 4
                                                                   chaves = N O
                                                                                  chaves = Q R S
                                                                                                    chaves = U V
                                  chaves = A B
                                                chaves = D E
                                                                                                                    chaves = Y
```

```
i = 1
                                                                                                                  t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                  excluirArvoreB(T,T.raiz,D)
       i = i+1
se i ≤ x.n e k == x.c<u>have[i</u>]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                  Raiz
                     se x.filhos[i+1].n ≥ t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                             filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                         n = 2
                                                                                                             n = 2
                                                                                                                             n = 2
                                                                                         n = 3
                                     n = 2
                                                    n = 4
                                                                         chaves = N O
                                                                                                             chaves = U V
                                     chaves = A B
                                                                                         chaves = Q R S
                                                    chaves = D E
                                                                                                                             chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                                  t=3
enquanto i \le x.n \in k > x.chave[i]
                                                                                                                  excluirArvoreB(T,T.raiz,D)
      i = i+1
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
             se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                 Raiz
                     se x.filhos[i+1].n ≥ t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                            filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                        n = 2
                                                                                        n = 3
                                                                                                            n = 2
                                                                                                                            n = 2
                                    n = 2
                                                    n = 4
                                                                        chaves = N 0
                                                                                                            chaves = U V
                                    chaves = A B
                                                                                         chaves = Q R S
                                                    chaves = D E
                                                                                                                            chaves = Y
```

```
t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                 excluirArvoreB(...,D)
       i = i+1
se i ≤ x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                 Raiz
                     se x.filhos[i+1].n ≥ t
                            //...
                     senão
                                                                           n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                            filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                                            n = 2
                                                                                                                            n = 2
                                    n = 2
                                                                        n = 2
                                                                                        n = 3
                                                    n = 4
                                                                        chaves = N 0
                                                                                                            chaves = U V
                                    chaves = A B
                                                                                         chaves = Q R S
                                                    chaves = D
                                                                                                                            chaves = \
```

i = 1

```
i = 1
                                                                                                                   t=3
enquanto i \le x.n \ e \ k > x.chave[i]
                                                                                                                   excluirArvoreB(...,D)
      i = i+1
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                  Raiz
                     se x.filhos[i+1].n ≥ t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                             chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                             filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                         n = 3
                                                                                                             n = 2
                                                                                                                             n = 2
                                                       n = 3
                                                                         n = 2
                                       n = 2
                                                                         chaves = N O
                                                                                                             chaves = U V
                                                                                          chaves = Q R S
                                       chaves = A B
                                                       chaves = | E
                                                                                                                              chaves = \
```

```
i = 1
                                                                                                                   t=3
enquanto i \le x.n \ e \ k > x.chave[i]
                                                                                                                   excluirArvoreB(...,D)
      i = i+1
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                  Raiz
                     se x.filhos[i+1].n ≥ t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                             chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                             filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                         n = 3
                                                                                                             n = 2
                                                                                                                             n = 2
                                                       n = 3
                                                                         n = 2
                                       n = 2
                                                                         chaves = N O
                                                                                                             chaves = U V
                                                                                          chaves = Q R S
                                       chaves = A B
                                                       chaves = | E
                                                                                                                              chaves = \
```

```
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T, x.filhos[i],k)
                                                                                   n = 3
                                                                                                                    n = 2
                                                                    n = 2
                                                                                                     n = 2
                                    n = 2
                                                   n = 3
                                                                                                     chaves = U V
                                                                    chaves = N O
                                                                                   chaves = Q R S
                                    chaves = A B
                                                   chaves = E
                                                                                                                    chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                      chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                T.raiz = x.filhos[1]
                                exclua o nodo x da memória
                                retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T, x.filhos[i],k)
                                                                                  n = 3
                                                                                                                   n = 2
                                                                   n = 2
                                                                                                    n = 2
                                    n = 2
                                                   n = 3
                                                                   chaves = N 0
                                                                                                    chaves = U V
                                                                                  chaves = Q R S
                                    chaves = A B
                                                   chaves = E | K
                                                                                                                    chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                       n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memória
                                 retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T, x.filhos[i],k)
                                                                   n = 2
                                                                                   n = 3
                                                                                                                    n = 2
                                                                                                     n = 2
                                    n = 2
                                                   n = 3
                                                                    chaves = N O
                                                                                                     chaves = U V
                                                                                   chaves = Q R S
                                    chaves = A B
                                                                                                                    chaves = Y
                                                   chaves = | E |
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                      chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                T.raiz = x.filhos[1]
                                exclua o nodo x da memoria
                                retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                                  n = 3
                                                                                                                    n = 2
                                                  n = 3
                                                                   n = 2
                                                                                                    n = 2
                                 n = 3
                                                                   chaves = N O
                                                                                                    chaves = U V
                                 chaves = A B
                                                                                  chaves = Q R S
                                                   chaves = E
                                                                                                                    chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                             t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                             excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
       //...
                                                           O processo é um pouco mais complicado. Note que o nodo b poderia
senão
      se x é folha
                                                           não ser folha, e precisariamos cuidar dos filhos dele também.
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                    b = irmaoImediatoComMaisChaves(x,i)
                    se b.n ≥ t
                                                                              Raiz
                           passar x.chaves[i] para x.filhos[i]
                           se b era irmão direito
                                  passar b.chaves[1] para x
                                                                         n = 3
                           senão
                                  passar b.chaves[b.n] para x
                                                                         chaves =
                    senão
                           merge (x.filhos[i], b, x.chaves[i])
                           excluir nodo b da memória
                                                                         filhos =
                           remova x.chaves[i] de x
                           se x.n < 1
                                 T.raiz = x.filhos[1]
                                 exclua o nodo x da memoria
                                 retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T, x.filhos[i],k)
                                                                     n = 2
                                                                                                                       n = 2
                                                    n = 2
                                                                                     n = 3
                                                                                                        n = 2
                                  n = 3
                                                                     chaves = N O
                                                                                                        chaves = U V
                                  chaves = A B
                                                                                     chaves = Q R S
                                                    chaves = | K
                                                                                                                       chaves = Y
```

```
função excluirArvoreB(T,x,k)
i = 1
                                                                                                          t=3
enquanto i \le x.n e k > x.chave[i]
                                                                                                          excluirArvoreB(T, T.raiz,B)
      i = i+1
se i \le x.n e k == x.chave[i]
      //...
senão
      se x é folha
             retorne //chave não encontrada
      senão
             se x.filhos[i].n < t</pre>
                   b = irmaoImediatoComMaisChaves(x,i)
                   se b.n ≥ t
                                                                            Raiz
                          passar x.chaves[i] para x.filhos[i]
                          se b era irmão direito
                                 passar b.chaves[1] para x
                                                                      n = 3
                          senão
                                 passar b.chaves[b.n] para x
                                                                       chaves =
                   senão
                          merge (x.filhos[i], b, x.chaves[i])
                          excluir nodo b da memória
                                                                       filhos =
                          remova x.chaves[i] de x
                          se x.n < 1
                                 T.raiz = x.filhos[1]
                                exclua o nodo x da memoria
                                 retorne excluirArvoreB(T,T.raiz,k)
retorne excluirArvoreB(T,x.filhos[i],k)
                                                                   n = 2
                                                                                                                    n = 2
                                                   n = 2
                                                                                   n = 3
                                                                                                     n = 2
                                 n = 3
                                                                    chaves = N O
                                                                                   chaves = Q R S
                                                                                                     chaves = U V
                                 chaves = A B
                                                                                                                    chaves = Y
                                                   chaves = | | | K |
```

```
i = 1
                                                                                                                 t=3
enquanto i ≤ x.n e k > x.chave[i]
                                                                                                                 excluirArvoreB(...,B)
       i = i+1
se i ≤ x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                 Raiz
                     se x.filhos[i+1].n \ge t
                            //...
                     senão
                                                                           n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                            filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                                                        n = 3
                                                                                                            n = 2
                                                                                                                            n = 2
                                                      n = 2
                                                                        n = 2
                                   n = 3
                                                                        chaves = N 0
                                                                                                            chaves = U V
                                                                                         chaves = Q R S
                                   chaves = A B
                                                       chaves = | K
                                                                                                                            chaves = \
```

funcão excluirArvoreB(T.x.k)

```
i = 1
                                                                                                                  t=3
enquanto i \le x.n \ e \ k > x.chave[i]
                                                                                                                  excluirArvoreB(...,B)
      i = i+1
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                  Raiz
                     se x.filhos[i+1].n \ge t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                             filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                       n = 2
                                                                                         n = 3
                                                                                                             n = 2
                                                                                                                             n = 2
                                       n = 2
                                                                         n = 2
                                                                                                             chaves = U V
                                                                         chaves = N O
                                                                                         chaves = Q R S
                                       chaves = A
                                                       chaves = | K
                                                                                                                             chaves = Y
```

```
i = 1
                                                                                                                  t=3
enquanto i \le x.n \ e \ k > x.chave[i]
                                                                                                                  excluirArvoreB(...,B)
      i = i+1
se i \le x.n e k == x.chave[i]
       se x é folha
              remova k de x
              retorne
       senão
              se x.filhos[i].n ≥ t
                     (p,j) = encontrarPred(x.filhos[i])
                     x.chave[i] = p.chaves[j]
                     retorne excluirArvoreB(T,p,p.chaves[j])
              senão
                                                                                  Raiz
                     se x.filhos[i+1].n \ge t
                            //...
                     senão
                                                                            n = 3
                            remova k de x
                            merge (x.filhos[i], x.filhos[i+1], k)
                            excluir nodo x.filhos[i+1] da memória
                                                                            chaves =
                            excluirArvoreB(x.filhos[i],k)
                            se x.n < 1
                                   T.raiz = x.filhos[1]
                                                                             filhos =
                                   exclua o nodo x da memória
                            retorne
senão
       se x é folha
              retorne
retorne excluirArvoreB(x.filhos[i],k)
                                                       n = 2
                                                                                         n = 3
                                                                                                             n = 2
                                                                                                                             n = 2
                                       n = 2
                                                                         n = 2
                                                                         chaves = N 0
                                                                                                             chaves = U V
                                                                                         chaves = Q R S
                                       chaves = A
                                                       chaves = | K
                                                                                                                             chaves = Y
```

Exercícios

- 1. Leia em Cormen et al. (2022) ou (2012) sobre o procedimento de exclusão de uma chave na Árvore B.
 - a. Compare a discussão apresentada com o algoritmo dado em aula.
 - b. Baseado na discussão do livro, e defina sua própria versão do algoritmo de exclusão.
- 2. Complete o algoritmo de exclusão, de forma que os nodos sejam carregados/descarregados nos pontos necessários.
- 3. Implemente os algoritmos em C. Assuma que todos os nodos estão na memória principal.

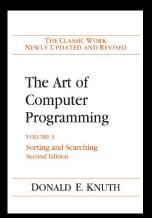
Referências

Estrutura de Dados e Algoritmos em C++. A. Drozdek. 4a ed. 2016.



C. Add INTRODUCTION TO ALL GORITHMS

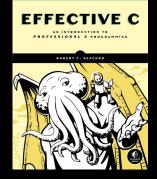
Knuth, D. The Art of Computer Programming: Volume 3: Sorting and Searching. 1998.



T. Cormen, C. Leiserson, R. Rivest, C. Stein. Algoritmos: Teoria e Prática. 4a ed. 2022.



Seacord, R. C. Effective C: An ntroduction to Professional C Programming. 2020.



Szwarcfiter, Markenzon, L. Estruturas de dados e seus algoritmos.2010.

Licença

Esta obra está licenciada com uma Licença Creative Commons Atribuição 4.0 Internacional.