

Estruturas de Dados

Cristina Gomes Fernandes

Árvores de busca binária (ABB)

ABB: árvore binária onde, para cada nó x , todo nó da subárvore esquerda de x tem *info* menor que $info(x)$, e todo nó da subárvore direita de x tem *info* maior que $info(x)$.

As operações de busca, inserção, remoção, mínimo, sucessor, etc, consome tempo, no pior caso, proporcional à altura da ABB.

A altura de uma ABB não balanceada por chegar a $n - 1$, onde n é o número de elementos armazenados na ABB. Ou seja, as operações acima, no pior caso, consomem tempo $\Theta(n)$.

Busca em ABBs

BUSQUE (T, x)

1 **se** $T = \text{NIL}$ **ou** $\text{info}(T) = x$

2 **então devolva** T

3 **senão se** $x < \text{info}(T)$

4 **então devolva** $\text{BUSQUE}(\text{esq}(T), x)$

5 **senão devolva** $\text{BUSQUE}(\text{dir}(T), x)$

Busca em ABBs

BUSQUE (T, x)

- 1 **se** $T = \text{NIL}$ **ou** $\text{info}(T) = x$
- 2 **então devolva** T
- 3 **senão se** $x < \text{info}(T)$
- 4 **então devolva** $\text{BUSQUE}(\text{esq}(T), x)$
- 5 **senão devolva** $\text{BUSQUE}(\text{dir}(T), x)$

Versão iterativa:

BUSQUE (T, x)

- 1 $p \leftarrow T$
- 2 **enquanto** $p \neq \text{NIL}$ **e** $\text{info}(p) \neq x$ **faça**
- 3 **se** $x < \text{info}(p)$
- 4 **então** $p \leftarrow \text{esq}(p)$
- 5 **senão** $p \leftarrow \text{dir}(p)$
- 6 **devolva** p

Inserção em ABBs

INSIRA (T, q) $\triangleright q$ é a célula nova

```
1   $p \leftarrow T$ 
2   $ant \leftarrow \text{NIL}$ 
3   $x \leftarrow \text{info}(q)$ 
4  enquanto  $p \neq \text{NIL}$  faça
5       $ant \leftarrow p$ 
6      se  $x < \text{info}(p)$ 
7          então  $p \leftarrow \text{esq}(p)$ 
8          senão  $p \leftarrow \text{dir}(p)$ 
9  se  $ant = \text{NIL}$ 
10     então  $T \leftarrow q$ 
11     senão se  $x < \text{info}(ant)$ 
12         então  $\text{esq}(ant) \leftarrow q$ 
13         senão  $\text{dir}(ant) \leftarrow q$ 
14   $\text{pai}(q) \leftarrow ant$ 
```

Versão recursiva da inserção

Nesta implementação, não temos o campo *pai*.

INSIRA (T, x)

1 $T \leftarrow \text{INSIRAREC}(T, x)$

INSIRAREC (T, x)

1 **se** $T = \text{NIL}$

2 **então** $q \leftarrow \text{NOVACÉLULA}(x, \text{NIL}, \text{NIL})$

3 **devolva** q

4 **se** $x < \text{info}(T)$

5 **então** $\text{esq}(T) \leftarrow \text{INSIRAREC}(\text{esq}(T), x)$

6 **senão** $\text{dir}(T) \leftarrow \text{INSIRAREC}(\text{dir}(T), x)$

7 **devolva** T

Versão recursiva da inserção

Nesta implementação, não temos o campo *pai*.

INSIRA (T, x)

1 $T \leftarrow \text{INSIRAREC}(T, x)$

INSIRAREC (T, x)

1 **se** $T = \text{NIL}$

2 **então** $q \leftarrow \text{NOVACÉLULA}(x, \text{NIL}, \text{NIL})$

3 **devolva** q

4 **se** $x < \text{info}(T)$

5 **então** $\text{esq}(T) \leftarrow \text{INSIRAREC}(\text{esq}(T), x)$

6 **senão** $\text{dir}(T) \leftarrow \text{INSIRAREC}(\text{dir}(T), x)$

7 **devolva** T

Exercício: Ajuste a rotina acima para que mantenha também o campo *pai* atualizado.

ABBs rubro-negras

Uma ABB é **rubro**-negra se

1. todo nó é **rubro** ou negro
2. toda folha (NIL) é negro
3. se um nó é **rubro**, então seus dois filhos são negros
4. todo caminho de um nó x até uma folha sua descendente tem o mesmo número de nós negros.

ABBs rubro-negras

Uma ABB é **rubro-negra** se

1. todo nó é **rubro** ou **negro**
2. toda folha (NIL) é **negra**
3. se um nó é **rubro**, então seus dois filhos são **negros**
4. todo caminho de um nó x até uma folha sua descendente tem o mesmo número de nós **negros**.

$rn(x)$: número de nós **negros** no caminho de um filho de x até uma folha descendente de x .

ABBs rubro-negras

Uma ABB é **rubro-negra** se

1. todo nó é **rubro** ou **negro**
2. toda folha (NIL) é **negra**
3. se um nó é **rubro**, então seus dois filhos são **negros**
4. todo caminho de um nó x até uma folha sua descendente tem o mesmo número de nós **negros**.

$rn(x)$: número de nós **negros** no caminho de um filho de x até uma folha descendente de x .

Lema: Uma ABB **rubro-negra** com n nós internos tem altura no máximo $2 \lg(n + 1)$.

rotações

O nó p é tal que $dir(p) \neq \text{NIL}$ e $cor(dir(p)) = \text{RUBRO}$.

ROTACIONEESQ (p)

- 1 $q \leftarrow dir(p)$
- 2 $dir(p) \leftarrow esq(q)$
- 3 $esq(q) \leftarrow p$
- 4 $cor(q) \leftarrow cor(p)$
- 5 $cor(p) \leftarrow \text{RUBRO}$
- 6 **devolva** q

rotações

O nó p é tal que $dir(p) \neq \text{NIL}$ e $cor(dir(p)) = \text{RUBRO}$.

ROTACIONEESQ (p)

- 1 $q \leftarrow dir(p)$
- 2 $dir(p) \leftarrow esq(q)$
- 3 $esq(q) \leftarrow p$
- 4 $cor(q) \leftarrow cor(p)$
- 5 $cor(p) \leftarrow \text{RUBRO}$
- 6 **devolva** q

Exercício: Escreva o ROTACIONEDIR.

Exercício: Ajuste estas rotinas para que mantenham o campo pai .

Ajusta cores

O nó p é interno.

TROQUECORES (p)

- 1 $cor(p) \leftarrow \text{OUTRACOR}(cor(p))$
- 2 $cor(esq(p)) \leftarrow \text{OUTRACOR}(cor(esq(p)))$
- 3 $cor(dir(p)) \leftarrow \text{OUTRACOR}(cor(dir(p)))$

Ajusta cores

O nó p é interno.

TROQUECORES (p)

- 1 $cor(p) \leftarrow \text{OUTRACOR}(cor(p))$
- 2 $cor(esq(p)) \leftarrow \text{OUTRACOR}(cor(esq(p)))$
- 3 $cor(dir(p)) \leftarrow \text{OUTRACOR}(cor(dir(p)))$

OUTRACOR (c)

- 1 **se** $c = \text{RUBRO}$
- 2 **então devolva** NEGRO
- 3 **senão devolva** RUBRO

Inserção em ABB rubro-negra

RUBRO (p)

1 **se** $p = \text{NIL}$

2 **então devolva** FALSO

3 **senão se** $\text{cor}(p) = \text{RUBRO}$

4 **então devolva** VERDADE

5 **senão devolva** FALSO

NEGRO (p)

1 **devolva** não **RUBRO**(p)

Inserção em ABB rubro-negra

RUBRO (p)

1 **se** $p = \text{NIL}$

2 **então devolva** FALSO

3 **senão se** $\text{cor}(p) = \text{RUBRO}$

4 **então devolva** VERDADE

5 **senão devolva** FALSO

NEGRO (p)

1 **devolva** não **RUBRO**(p)

INSIRA (T, x)

1 $T \leftarrow \text{INSIRAREC}(T, x)$

2 $\text{cor}(T) \leftarrow \text{NEGRO}$ ▷ a raiz é sempre negra

Inserção em ABB rubro-negra

Esta é a inserção para **árvores 2-3**:

INSIRAREC (T, x)

- 1 **se** $T = \text{NIL}$
- 2 **então** $q \leftarrow \text{NOVACÉLULA}(x, \text{NIL}, \text{NIL}, \text{RUBRO})$
- 3 **devolva** q
- 4 **se** $x < \text{info}(T)$
- 5 **então** $\text{esq}(T) \leftarrow \text{INSIRAREC}(\text{esq}(T), x)$
- 6 **senão** $\text{dir}(T) \leftarrow \text{INSIRAREC}(\text{dir}(T), x)$
- 7 **se** $\text{RUBRO}(\text{dir}(T))$ e $\text{NEGRO}(\text{esq}(T))$
- 8 **então** $T \leftarrow \text{ROTACIONEESQ}(T)$
- 9 **se** $\text{RUBRO}(\text{esq}(T))$ e $\text{RUBRO}(\text{esq}(\text{esq}(T)))$
- 10 **então** $T \leftarrow \text{ROTACIONEDIR}(T)$
- 11 **se** $\text{RUBRO}(\text{esq}(T))$ e $\text{RUBRO}(\text{dir}(T))$
- 12 **então** $\text{TROQUECORES}(T)$
- 13 **devolva** T

Inserção em ABB rubro-negra

Esta é a inserção para **árvores 2-3-4**:

INSIRAREC (T, x)

```
1  se  $T = \text{NIL}$ 
2    então  $q \leftarrow \text{NOVACÉLULA}(x, \text{NIL}, \text{NIL}, \text{RUBRO})$ 
3    devolva  $q$ 
4  se  $\text{RUBRO}(\text{esq}(T))$  e  $\text{RUBRO}(\text{dir}(T))$ 
5    então  $\text{TROQUECORES}(T)$ 
6  se  $x < \text{info}(T)$ 
7    então  $\text{esq}(T) \leftarrow \text{INSIRAREC}(\text{esq}(T), x)$ 
8    senão  $\text{dir}(T) \leftarrow \text{INSIRAREC}(\text{dir}(T), x)$ 
9  se  $\text{RUBRO}(\text{dir}(T))$  e  $\text{NEGRO}(\text{esq}(T))$ 
10  então  $T \leftarrow \text{ROTACIONEESQ}(T)$ 
11  se  $\text{RUBRO}(\text{esq}(T))$  e  $\text{RUBRO}(\text{esq}(\text{esq}(T)))$ 
12  então  $T \leftarrow \text{ROTACIONEDIR}(T)$ 
13  devolva  $T$ 
```

Exercício

Simule a inserção em uma árvore rubro-negra inicialmente vazia das seguintes chaves:

20, 17, 38, 40, 53, 10, 6, 16, 23, 14, 11, 50, 45.

Faça a simulação duas vezes: uma com o INSIRA de árvore 2-3 e outra com o INSIRA de árvore 2-3-4.

Analise e discuta as diferenças que você observa.