

Árvore Preto-Vermelho

Rohit Gheyi
rohit@dsc.ufcg.edu.br

1

Introdução

- Rudolf Bayer: "Symmetric Binary-Trees: Data Structures and Maintenance Algorithms." *Acta Informat.* **1**, 290-306, 1972.
 - O nome atual em 1978 por Leonidas J. Guibas e Robert Sedgwick
- Estrutura de dados que **se mantém balanceada**
- **Eficiente** na prática



2

Características

- Árvore Binária de Pesquisa
- Cada nó possui uma cor
 - Preto
 - Vermelho
- Nenhuma altura é maior do que o dobro da outra (razoavelmente balanceada)
- Objetivo
 - Remover o problema do pior caso onde a altura é $O(n)$

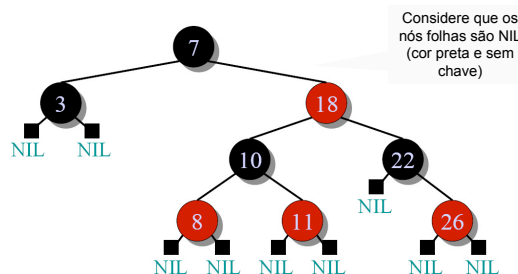
3

Invariante

1. Um nó é **preto** ou **vermelho**
2. Os **nós folha** são **pretos**
3. Os **filhos** de um nó **vermelho** são **pretos**
4. Qualquer caminho descendente de um nó até um nó folha possui o mesmo número de nós pretos (nós que possuem chaves)
 - Black-height(x)
 - Começa a contar a partir do filho
5. A raiz é preta

4

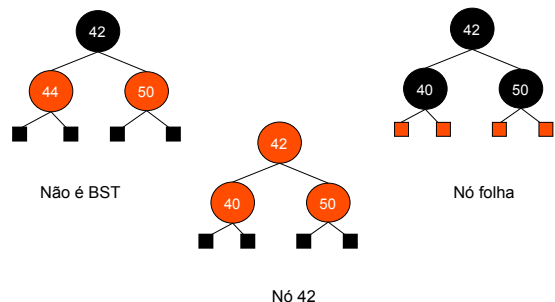
Exemplo



5

Exercício 1

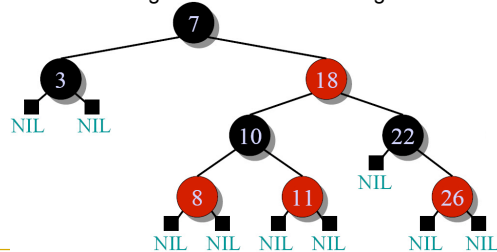
- Quais destas árvores são preto-vermelho?



6

Exercício 2

- Qual o black-height do nó 7?
 - 2
 - O black-height da raiz é o black height da árvore pv



7

Altura

- Uma árvore preto-vermelho com n nós internos tem no máximo a altura de:

$$h \leq 2 \cdot \lg(n+1)$$

- Intuição, junte os nós vermelhos no pretos
- Prova por indução
 - Assuma que o número de nós internos é $2^{bh(x)} - 1$

8

Nodo

```
class RBNode {
    RBNode left, right, parent;
    int key;
    Object satellite;
    boolean ehPreto; ...
}
```

9

Exercício

- Como é feita a pesquisa em uma árvore preto-vermelho?

10

Pesquisa

- A pesquisa realizada em uma árvore preto-vermelho é igual a realizada em uma árvore binária de pesquisa

11

Algoritmo

- Faça a análise do algoritmo.

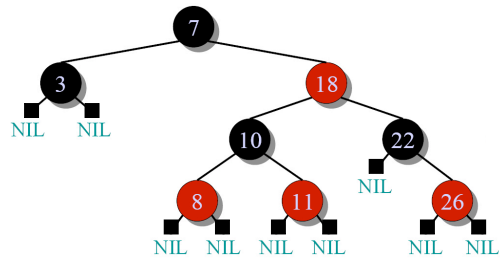
Lembrar que é $O(h)$. Mas $h = \log n$. Logo é $O(\log n)$

```
public RBNode TREE-SEARCH(int key) {
    RBNode t = this.root;
    while( t != null ) {
        if (key < t.element)
            t = t.left;
        else if (key > t.element)
            t = t.right;
        else return t;
    }
    return null;
}
```

12

Exercício 1

- Pesquise os nodos 8 e 26?



13

Exercício 2

- Como inserir em uma árvore preto-vermelho?
É só inserir no como no TREE-INSERT?

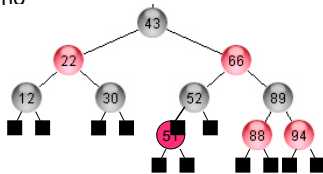
```

TREE-INSERT (T, z)
y ← NIL
x ← root [T]
while x ≠ NIL do
y ← x
if key [z] < key[x]
then x ← left[x]
else x ← right[x]
p[z] ← y
if y = NIL
then root [T] ← z
else if key [z] < key [y]
then left [y] ← z
else right [y] ← z
    
```

14

Exercício 1

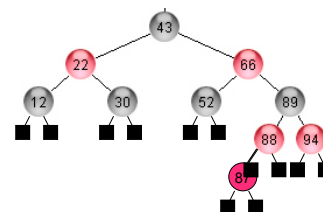
- Insira o nó 51. Qual a cor do novo nó?
Justifique. Assuma que a árvore possui nós
folhas pretos e sem chaves.
- Vermelho



15

Exercício 2

- Insira o nó 87. Preserva o invariante?
- A propriedade 3 não é satisfeita
 - O filho do nó 88 não é preto



16

Problema

Como **manter o invariante** de uma
árvore preto-vermelho após uma inclusão?

17

Inserir

- Insere o nodo normalmente com o **TREE-INSERT(T,n)**
 - Considere n como nó vermelho
- Se o pai de n é vermelho
 - Quebra o invariante 3
 - Os filhos de um nó vermelho são pretos
 - Aplica procedimento para ajustar invariante
 - Casos 1, 2 e 3
- Sempre **foque no tio** do nó n

18

Algoritmo

```

RB-INSERT(T, z)
  TREE-INSERT(T, z)
  color[z] ← RED
  RB-INSERT-FIXUP(T, z)
    
```

19

Caso 1: Inserção

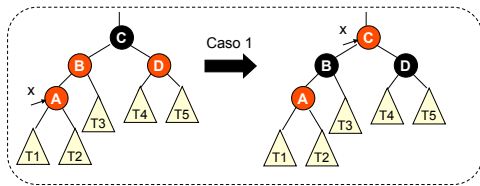
- Seja n o nó com problema
 - Ele e o pai são vermelhos
- Condição:
 - Tio (D) é **vermelho**
- Resultado:
 - Troca as cores do avô (C), do pai (B) e do tio (D)



20

Exercício

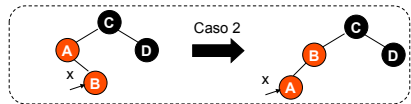
- Análise a black-height dos ramos e T1-T5 antes e depois da transformação.



21

Caso 2: Inserção

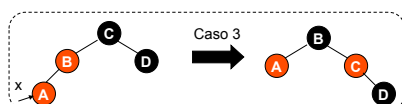
- Condição:
 - Tio (D) é **preto**
 - O nó B, seu pai (A) e seu avô (C) formam um **joelho**
- Resultado:
 - Rotaciona o pai (A) para a esquerda ou direita
- Pós condição:
 - O resultado do caso 2 cai sempre no **caso 3**



22

Caso 3: Inserção

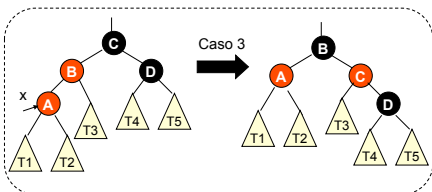
- Condição:
 - Tio (D) é **preto**
 - O nó A, seu pai (B) e seu avô (C) estão em uma "reta"
- Resultado:
 - Rotaciona o avô (C) para a esquerda ou direita
 - Troca a cor do avô (C) e pai (B)



23

Exercício

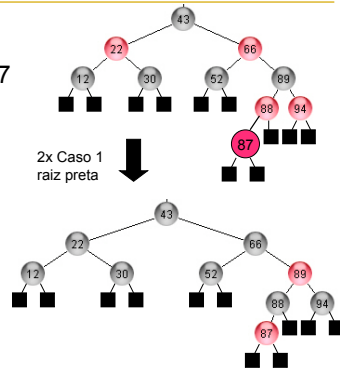
- Análise a black-height dos ramos e T1-T5 antes e depois da transformação.



24

Exemplo

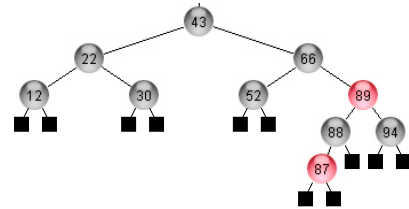
- Insira o nodo 87



25

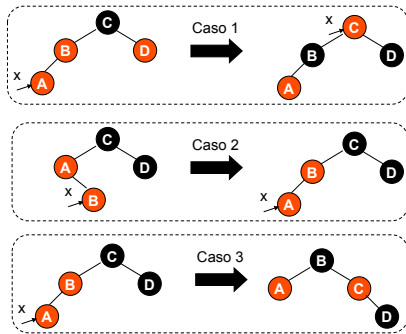
Exercício

- Insira os nodos 83, 70, 34, 71.



26

Casos



27