

Alberi - Lezione 12 FERONE

• VISITA DI ALBERI BINARI

1) VISITA-ANTICIPATA (x)

if $x \neq \text{NIL}$

visita (x)

VISITA-ANTICIPATA (Left [x])

VISITA-ANTICIPATA (Right [x])

2) VISITA-SIMMETRICA \rightarrow INORDER-TREE-WALK (x)

if $x \neq \text{NIL}$

INORDER-TREE-WALK (Left [x])

visita (x)

INORDER-TREE-WALK (Right [x])

3) VISITA-DIFFERITA (x)

if $x \neq \text{NIL}$

VISITA-DIFFERITA (Left [x])

VISITA-DIFFERITA (Right [x])

visita (x)

2. RICERCA DI UN NODO CON UNA DATA CHIAVE IN ABR

1) TREE-SEARCH (x, k)
 ↑ puntatore alla radice dell'albero

```
if  $x == \text{NIL}$  ||  $k == \text{key}[x]$ 
    return  $x$ 
if  $k < \text{key}[x]$ 
    return TREE-SEARCH (Left [ $x$ ],  $k$ )
else
    return TREE-SEARCH (Right [ $x$ ],  $k$ )
```

2) TREE-ITERATIVE-SEARCH (x, k)

```
while  $x \neq \text{NIL}$  AND  $k \neq \text{key}[x]$ 
    if  $k < \text{key}[x]$ 
         $x = \text{Left}[x]$ 
    else
         $x = \text{Right}[x]$ 
return  $x$ 
```

3. MININO, MASSIMO, SUCCESSORE, PREDECESSORE

1) TREE-MINIMUM (x)

while $\text{Left}[x] \neq \text{NIL}$

do $x \leftarrow \text{Left}[x]$

return x

2) TREE-MAXIMUM (x)

while $\text{Right}[x] \neq \text{NIL}$

do $x \leftarrow \text{Right}[x]$

return x

3) TREE-SUCCESSOR (x)

if $\text{Right}[x] \neq \text{NIL}$

return TREE-MINIMUM ($\text{Right}[x]$)

$y \leftarrow p[x]$

while $y \neq \text{NIL}$ AND $x == \text{Right}[y]$

$x \leftarrow y$

$y \leftarrow p[y]$

return y

4) TREE-PREDECESSOR (x)

if $\text{Left}[x] \neq \text{NIL}$

 return TREE-MAXIMUM ($\text{Left}[x]$)

$y \leftarrow p[x]$

while $y \neq \text{NIL}$ AND $x = \text{Left}[y]$

$x \leftarrow y$

$y \leftarrow p[y]$

return y

4) INSERIMENTO E CANCELLAZIONE IN ABR

1) TREE-INSERT (T, z)

$y \leftarrow \text{NIL}$

$x \leftarrow \text{root}[T]$

while $x \neq \text{NIL}$

$y \leftarrow x$

if $\text{key}[z] < \text{key}[x]$

$x \leftarrow \text{Left}[x]$

else

$x \leftarrow \text{Right}[x]$

$p[z] \leftarrow y$

if $y = \text{NIL}$

$\text{root}[T] \leftarrow z$

else if $\text{key}[z] < \text{key}[y]$

$\text{Left}[y] \leftarrow z$

else

$\text{Right}[y] \leftarrow z$

2) TRANSPLANT (T, u, v) per ABR

if $p[u] == \text{NIL}$
 $\text{root}[T] = v$
else if $u == \text{left}[p[u]]$
 $\text{left}[p[u]] = v$
else
 $\text{right}[p[u]] = v$
if $v \neq \text{NIL}$
 $p[v] = p[u]$

3) TREE-DELETE (T, z)

if $\text{left}[z] == \text{NIL}$
 $\text{TRANSPLANT}(T, z, \text{right}[z])$
else if $\text{right}[z] == \text{NIL}$
 $\text{TRANSPLANT}(T, z, \text{left}[z])$
else
 $y = \text{TREE-MINIMUM}(\text{right}[z])$
if $p[y] \neq z$
 $\text{TRANSPLANT}(T, y, \text{right}[y])$
 $\text{right}[y] = \text{right}[z]$
 $p[\text{right}[y]] = y$

$\text{TRANSPLANT}(T, z, y)$

$\text{left}[y] = \text{left}[z]$
 $p[\text{left}[y]] = y$

ALBERI RED & BLACK - LEZIONE 13 FERONE

• ROTAZIONI

1) LEFT-ROTATE (T, x)

$y \leftarrow \text{right}[x]$

$\text{right}[x] \leftarrow \text{left}[y]$

if $\text{left}[y] \neq \text{NIL}$

$p[\text{left}[y]] \leftarrow x$

$p[y] \leftarrow p[x]$

if $p[x] == \text{NIL}$

$\text{root}[T] \leftarrow y$

else if $x == \text{left}[p[x]]$

$\text{left}[p[x]] \leftarrow y$

else

$\text{right}[p[x]] \leftarrow y$

$\text{left}[y] \leftarrow x$

$p[x] \leftarrow y$

2) RIGHT-ROTATE (T, x) \rightarrow uguale invertendo right e left
nel codice sopra

2. INSERIMENTO IN ALBERI R&B : RB_INSERT

TREE_INSERT (T, x)

color [x] \leftarrow RED

while $x \neq \text{root}[T]$ and color [$p[x]$] = RED

 olo if $p[x] = \text{left}[p[p[x]]]$

 then $y \leftarrow \text{right}[p[p[x]]]$

 if color [y] = RED

 then color [$p[x]$] \leftarrow BLACK

 color [y] \leftarrow BLACK

 color [$p[p[x]]$] \leftarrow RED

$x \leftarrow [p[p[x]]]$

 else if $x = \text{right}[p[x]]$

 then $x \leftarrow [p[x]]$

 LEFT_ROTATE (T, x)

 color [$p[x]$] \leftarrow BLACK

 color [$p[p[x]]$] \leftarrow RED

 RIGHT_ROTATE ($T, p[p[x]]$)

else (uguale al then con RIGHT e LEFT scambiati)

color [$\text{root}[T]$] \leftarrow BLACK

3 CANCELLAZIONE IN ALBERI R & B

1) RB-TRANSPLANT (T, u, v)

if $p[u] == NIL[T]$

$\text{root}[T] = v$

else if $u == \text{left}[p[u]]$

$\text{left}[p[u]] = v$

else

$\text{right}[p[u]] = v$

$p[v] = p[u]$

2) RB-DELETE (T, z)

$y = z$

$y\text{-orig-color} = \text{color}[y]$

if $\text{left}[z] == \text{NIL}[T]$

$x = \text{right}[z]$

$\text{RB-TRANSPLANT}(T, z, \text{right}[z])$

else if $\text{right}[z] == \text{NIL}[T]$

$x = \text{left}[z]$

$\text{RB-TRANSPLANT}(T, z, \text{left}[z])$

else

$y = \text{TREE-MINIMUM}(\text{right}[z])$

$y\text{-orig-color} = \text{color}[y]$

$x = \text{right}[y]$

if $p[y] == z$

$p[x] = y \quad // \text{se } x == \text{NIL}[T]$

else

$\text{RB-TRANSPLANT}(T, y, \text{right}[y])$

$\text{right}[y] = \text{right}[z]$

$p[\text{right}[y]] = y$

$\text{TRANSPLANT}(T, z, y)$

$\text{left}[y] = \text{left}[z]$

$p[\text{left}[y]] = y$

$\text{color}[y] = \text{color}[z] \quad // y \text{ prende il colore di } z$

if $y\text{-orig-color} == \text{BLACK}$

$\text{RB-DELETE-FIX-UP}(T, x)$

3) RB-DELETE-FIXUP (T, x)

RB_DELETE_FIXUP(T, x)

```

while  $x \neq root[T]$  and  $color[x] = BLACK$ 
    do if  $x = left[p[x]]$ 
        then  $w \leftarrow right[p[x]]$ 
            if  $color[w] = RED$ 
                then  $color[w] \leftarrow BLACK$ 
                     $color[p[x]] \leftarrow RED$ 
                    LEFT_ROTATE( $T, p[x]$ )
                     $w \leftarrow right[p[x]]$ 
            if  $color[left[w]] = BLACK$  and  $color[right[w]] = BLACK$ 
                then  $color[w] \leftarrow RED$ 
                     $x \leftarrow p[x]$ 
                else if  $color[right[w]] = BLACK$ 
                    then  $color[left[w]] \leftarrow BLACK$ 
                         $color[w] \leftarrow RED$ 
                        RIGHT_ROTATE( $T, w$ )
                         $w \leftarrow right[p[x]]$ 
                     $color[w] \leftarrow color[p[x]]$ 
                     $color[p[x]] \leftarrow BLACK$ 
                     $color[right[w]] \leftarrow BLACK$ 
                    LEFT_ROTATE( $T, p[x]$ )
                     $x \leftarrow root[T]$ 
    else (analogo al ramo then con right e left scambiati)
     $color[x] \leftarrow BLACK$ 

```

