SOLUZIONI APPELLO

ALGORITMI 15/07

DOMANDE A, B, C

A) Si deve dimostrare che:

1 - $T(n) \leq c \cdot n \qquad c > 0$

2 - $T(n) \geq d \cdot n \qquad d > 0 \qquad \forall n \geq n_0$

1 - $T(n) \leq c\left(\frac{n}{2}\right) + c\left(\frac{n}{4}\right) + n =$

$= n\left(\frac{c}{2} + \frac{c}{4} + 1\right) = n\left(\frac{3}{4}c + 1\right)$

$\leq c \cdot n \iff c \geq \frac{3}{4}c + 1 \iff$

$\boxed{c \geq 4} \quad (n \geq 0)$

2 - Analogo al punto 1, ma con segno cambiato:

$\boxed{d \leq 4}$

Poichè $T(n) = O(n)$ ①

e $T(n) = \Omega(n)$ ②

$\Longrightarrow T(n) = \Theta(n)$.

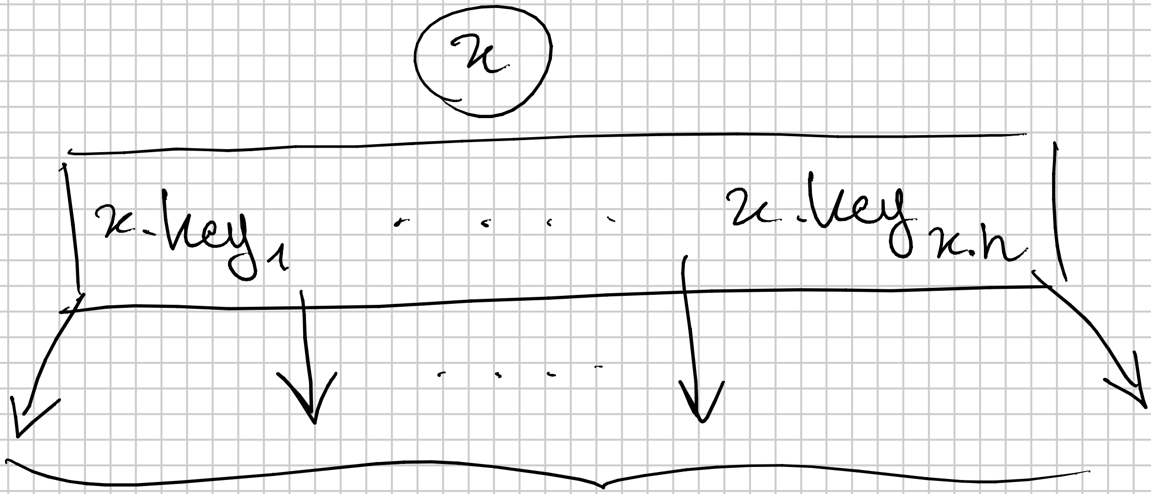
$(\Longleftarrow)$

B) Def. B-ALBERO

Sia T un B-albero

- T.root è la radice
- x è un nodo qualsiasi

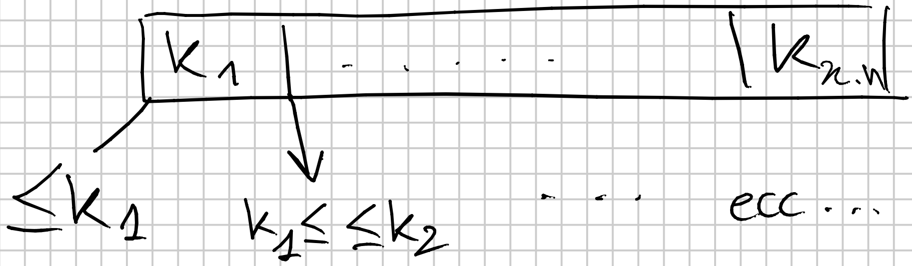
$\forall x,\ x.n = \#$ chiavi



$\forall x,\ \exists\ (x.n+1)$ figli

se $x.leaf = FALSE$

Le chiavi sono ordinate.



Le foglie di un B-albero hanno tutte la stessa profondità

Sia t il grado minimo del B-albero.
Allora:

$$\forall x,\ x.n \geq t-1 \quad (x \neq T.root)$$
$$\underline{e} \quad x.n \leq 2t-1$$

Altezza minima B-albero con grado t e n chiavi?

⇝ Altezza minima: ogni nodo ha il massimo numero di chiavi $(2t-1)$
$\Rightarrow$ ha $(2t-1)+1$ figli $= 2t$ se non foglia.

$$n = \#\text{nodi}$$
$$\overset{!}{=} (2t-1) + 2t(2t-1) + (2t)^2(2t-1) + \dots + (2t)^h(2t-1)$$
$$= (2t-1)\sum_{i=0}^{h}(2t)^i$$
$$\overset{\circledast}{=} \cancel{(2t-1)}\ \frac{2t^{(h+1)}-1}{\cancel{(2t-1)}} =$$

$$= 2t^{(h+1)} - 1 .$$

$$n = 2t^{(h+1)} - 1 \iff$$

$$n + 1 = 2t^{(h+1)} \iff$$

$$\log_{2t}(n+1) = h + 1 \iff$$

$$h = \boxed{\log_{2t}(n+1) - 1} .$$

NOTA: se $x \neq 1$

⊛

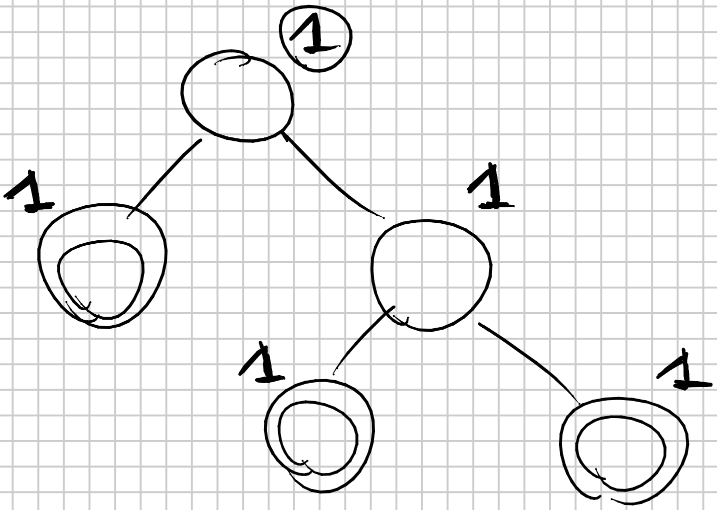
$$\sum_{n=0}^{N} x^n = \frac{x^{(N+1)} - 1}{x - 1} .$$

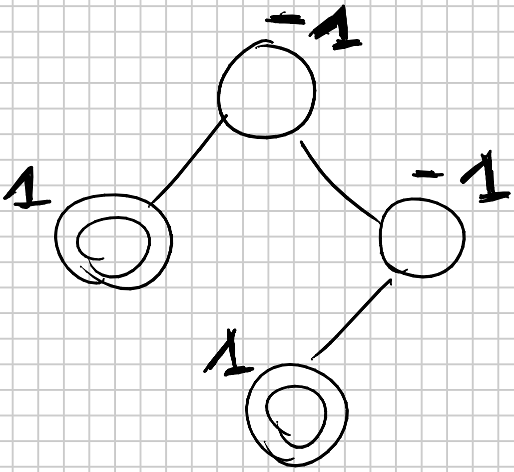
c) $\forall$ nodo x , $x.col = \begin{cases} B \\ R \end{cases}$

```
bh(x)
  if(x=NIL)
      return 0
  else
      countl = bh(x.left)
      countr = bh(x.right)
      if(countl = -1 OR countr = -1
            OR (countl ≠ countr AND
                 countl, countr ≠ 0)))   (*)
              return -1
      else
            count = max{countl, countr}
            if(x.color = B)
                  return count + 1
            else
                  return count
```

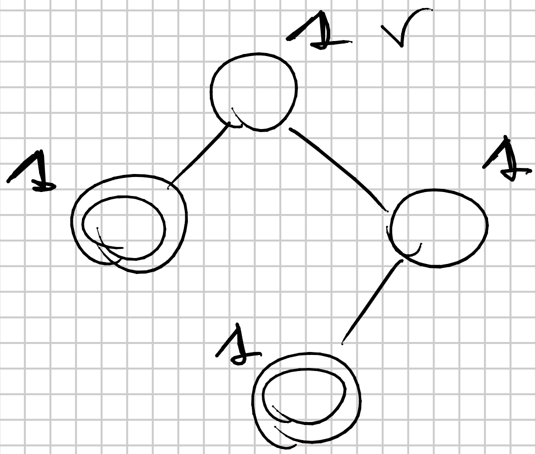
* Durante la correzione la mancanza di questa condizione non è stata considerata un errore grave.
Serve a gestire il caso in cui manchi uno dei due figli in un nodo x.

Es.





senza condizione *



con condizione *

Appunti dalla correzione del 24/07