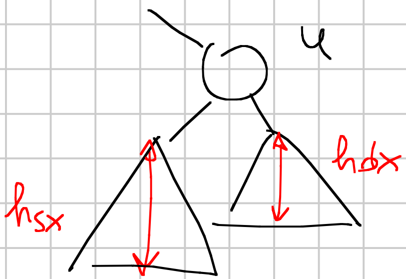
Alberi AVL

Adel'son - Velsky e Landis, 1962

Sono ABR 1-bilanciati: $\forall$ nodo u dell'albero



T(u) = sottoalbero di radice u

T(u.left) = sottoalbero sx di u

T(u.right) = sottoalbero dx di u

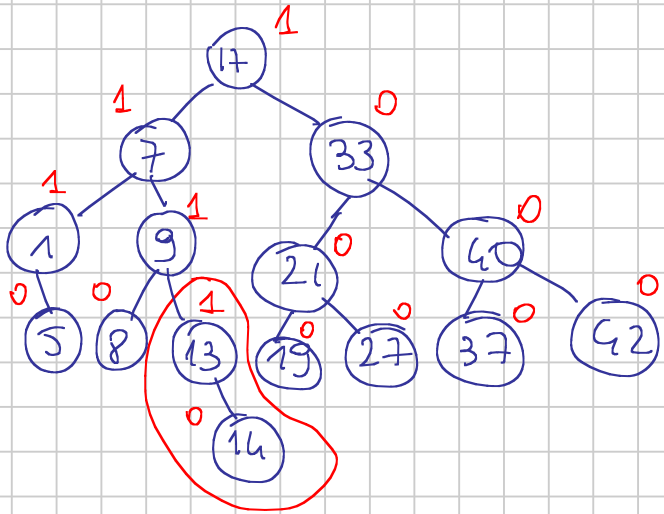
hsx = altezza di T(u.left)

hdx = altezza di T(u.right)

$$\boxed{\Delta = |hsx - hdx| \leq 1}$$

FATTORE DI BILANCIAMENTO del nodo u

ESEMPIO



$\forall$ foglia, $\Delta = 0$

1-bilanciato

OBIETTIVO: dimostrare che $\forall$ albero 1-bilanciato di n nodi

$$h = O(\log n)$$

↳ dimostreremo che $\forall$ albero 1-bilanciato di altezza h, $\boxed{n \geq c^h}$ $c > 1$ costante.

↳ $\left[\log_c n \geq h \Rightarrow h = O(\log n) \right]$

Titolo nota 26/02/2016

Alberi 1-bilanciati MINIMALI → Alberi di Fibonacci Fib_h

↳ raggiungono altezza h con il minor numero possibile di nodi

⇓

se rimuovo un nodo, l'albero ottenuto NON è più un albero 1-bilanciato di altezza h.

$\boxed{ABCB}$: tutti i nodi hanno $\Delta = 0$

$\boxed{Fib_h}$: tutti i nodi interni hanno $\Delta = 1$

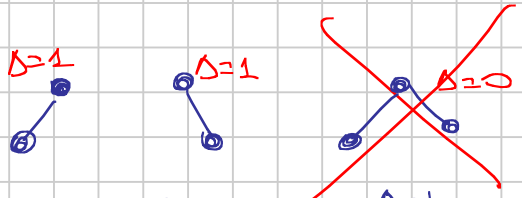
Alberi di Fibonacci

n_h = numero di nodi di Fib_h.

Fib_0 $h=0$

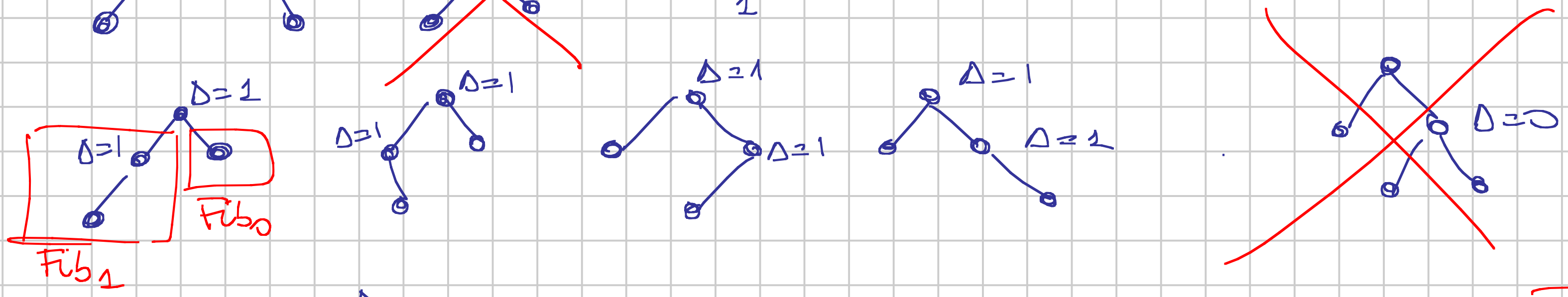
$n_0 = 1$

Fib_1 $h=1$



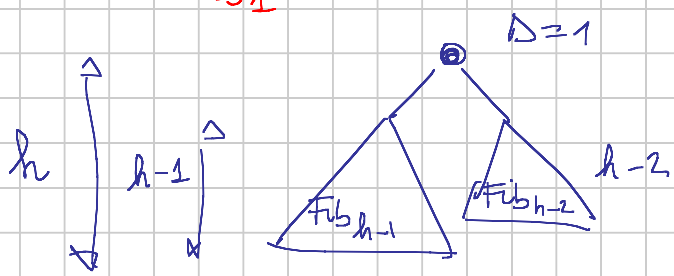
$n_1 = 2$

Fib_2 $h=2$

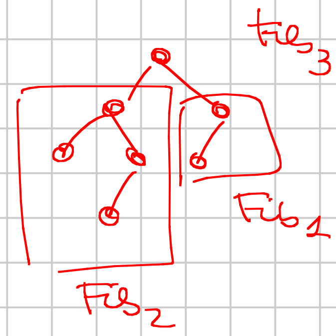


$n_2 = 4$

Fib_h



$$\boxed{n_h = \underbrace{1}_{\text{radice}} + n_{h-1} + n_{h-2}}$$

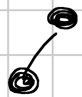


LEMMA

Fib_h è un albero 1-bilanciato di altezza h minimale
(cioè con il minor numero possibile di nodi)

DIM Per induzione su h.

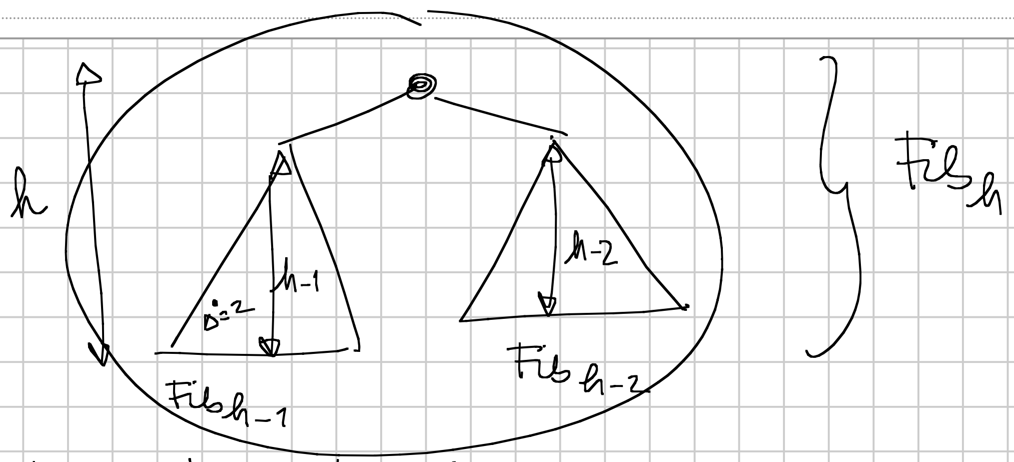
BASE Fib_0  se rimuovo l'unico nodo, l'altezza $\downarrow$ (da 0 a -1)

Fib_1  rimuovendo uno dei due nodi, $h \searrow$ (da 1 a 0)

IPOTESI INDUTTIVA $\forall\, l < h$ Fib_l sia un albero 1-bilanciato di altezza l minimale

Titolo nota 26/02/2016

PASSO



X assurdo: Fib_h NON ~~sta~~ SIA MINIMALE

OSS. 1) non posso rimuovere la radice
$\Rightarrow$ il nodo da rimuovere deve $\in Fib_{h-1}$ oppure a Fib_{h-2}

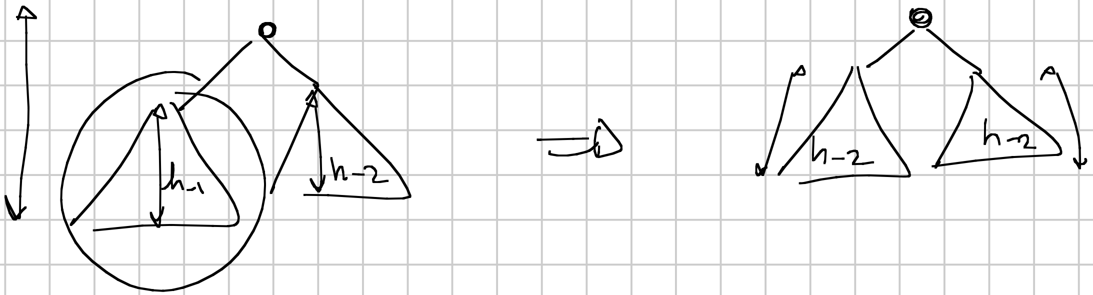
minimali per ipotesi induttiva

2) il nodo che si rimuove NON può sbilanciare il sottoalbero da cui è rimosso ($\Delta: 1 \to 2$)
perché ~~stesso~~ anche Fib_h perderebbe la proprietà di essere 1-bilanciato

$\Rightarrow$ il nodo rimosso (u) da uno dei due sottoalberi dell' ↘ altezza

2 casi

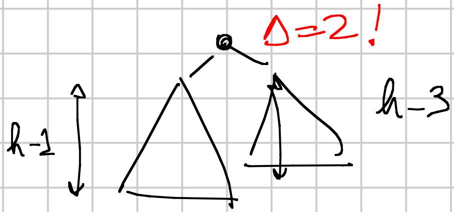
1) $u \in Fib_{h-1}$ $\Rightarrow$ dopo la rimozione di u
Fib_{h-1} avrà altezza $h-2$



$h \rightarrow h-1$

$\Rightarrow$ l'altezza di Fib_h diventa $h-1$

2) $u \in Fib_{h-2}$



$\Rightarrow$ l'altezza di Fib_{h-2} diventa $h-3$

e il Δ della radice diventa 2

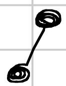
LEMMA $\boxed{n_h = F_{h+3} - 1}$

F_{h+3} = h-3-esimo numero di Fibonacci

F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	$\dots$
0	1	1	2	3	5	8	13	$\dots$

DIM per induzione su h.

BASE $h=0$ $\quad Fib_0$  $\quad n_0 = 1 = F_{0+3} - 1 = 2-1$ ✓

$h=1$ $\quad Fib_1$  $\quad n_1 = 2$

$F_{1+3} - 1 = F_4 - 1 = 2$ ✓

i.i. $\forall\, l < h \quad n_l = F_{l+3} - 1.$

PASSO Fib_h

$$n_h = 1 + n_{h-1} + n_{h-2} \overset{i.i}{=} \cancel{1} + \left(F_{(h-1)+3} - \cancel{1}\right) + \left(F_{(h-2)+3} - 1\right) =$$

$$= F_{h+2} + F_{h+1} - 1 = F_{h+3} - 1 \qquad \square$$

$$\boxed{F_n = \frac{\phi^n}{\sqrt{5}} - \frac{\hat{\phi}^n}{\sqrt{5}}} \to 0$$

$$\phi = \frac{1+\sqrt{5}}{2}$$

$$\hat{\phi} = \frac{1-\sqrt{5}}{2}$$

$$\boxed{x^2 - x - 1 = 0}$$

$$x_1 = \phi \qquad x_2 = \hat{\phi}$$

$$\phi \cong 1.618\ldots$$

$$\hat{\phi} = -0.618\ldots$$

$$\Rightarrow \boxed{\phi^2 = \phi + 1}$$

$$F_n > \frac{\phi^{n-1}}{\sqrt{5}}$$

crescita esponenziale

LEMMA ϕ $\quad n_h \geq \phi^h \quad \forall h \geq 0$

$\phi = 1.618\ldots > 1$

DIM. Per induzione su h.

BASE $\boxed{h=0}$

$n_0 = 1$

$\phi^0 = 1$

$n_0 = \phi^0$ ✓

$\boxed{h=1}$

$n_1 = 2$

$\phi^1 = 1.618\ldots < 2 \Rightarrow n_1 > \phi^1$ ✓

i.i. $\forall \ell < h \quad n_\ell \geq \phi^\ell$

PASSO

$$n_h = 1 + n_{h-1} + n_{h-2} > n_{h-1} + n_{h-2}$$

$$\underset{i.i}{\geq} \phi^{h-1} + \phi^{h-2} = \phi^{h-2}(\phi + 1) = \phi^{h-2} \cdot \phi^2 = \phi^h$$

$\phi^2 = \phi + 1$

$\square$

~~TEOR~~

TEOREMA

Ogni albero 1-bilanciato con n nodi ha altezza $h = O(\log n)$.

DIM

Sia h l'altezza dell'albero

$$\underbrace{n \geq n_h}_{\text{x la minimalità degli Alberi di Fibonacci}} \underbrace{\geq c^h}_{\text{lemma } \phi} \qquad [c > 1,\ c = \phi]$$

$$\Rightarrow \underbrace{n \geq c^h}_{\Omega} \Rightarrow \underbrace{\log_c n \geq h}_{O} \Rightarrow \boxed{h = O(\log n)}$$

$\square$

Corollario Ogni albero AVL ha altezza $O(\log n)$.

$\square$

AVL: implementazione di operazioni

INTERROGAZIONI (come per ABR)

$$\boxed{\underbrace{T(n) = O(h)}_{ABR} = O(\log n)}$$

Operazioni di modifica

INSERIMENTO : aggiungo una foglia (f)

più costoso ↓ sbilanciamento ↓

Nodo critico: ~~xxx~~ antenato ~~xxx~~ di (f), più profondo nell'albero (più vicino a f) il cui Δ è diventato 2.

x l'implementazione nodo x

x.key
x.dati
x.p
x.left
x.right
x.altezza → altezza di $T(x)$

Inserimento $T(n) = O(\log n)$

→ si cerca il nodo che diventerà padre della nuova foglia f

↳ costo $O(h) = O(\log n)$

→ si cerca il nodo critico $O(h) = O(\log n)$

→ si bilancia il nodo critico con una rotazione semplice o doppia $\Theta(1)$

$T(n) = O(\log n)$

CANCELLAZIONE :

logica (si marcano i nodi "cancellati")

- algoritmo ABR $O(h) = O(\log n)$
- si cerca il nodo critico $O(h)$
- si ribilancia il nodo critico $\Theta(1)$
- ~~deve~~ può succedere di dover eseguire rotazioni su tutti gli antenati del nodo critico

CASO PESSIMO : $\Theta(\log n)$ rotazioni

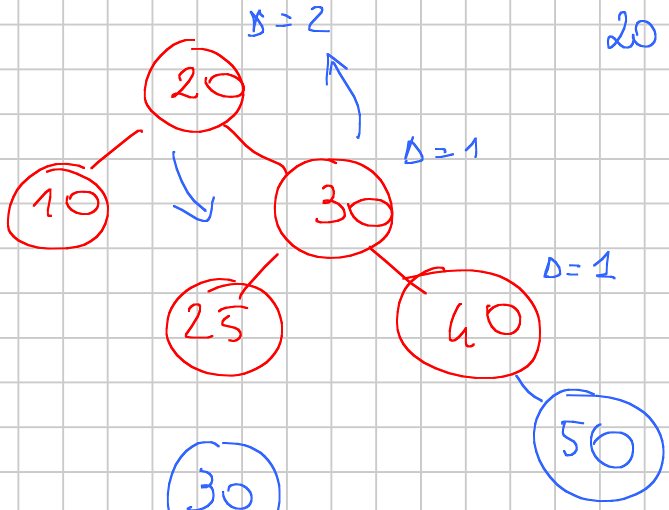
$$T(n) = \underbrace{O(\log n)}_{\text{cancellare}} + \underbrace{O(\log n) \cdot \Theta(1)}_{\text{ripristinare il bilanciamento}} = O(\log n)$$

rotazioni

costo costante rotazioni

ESEMPIO

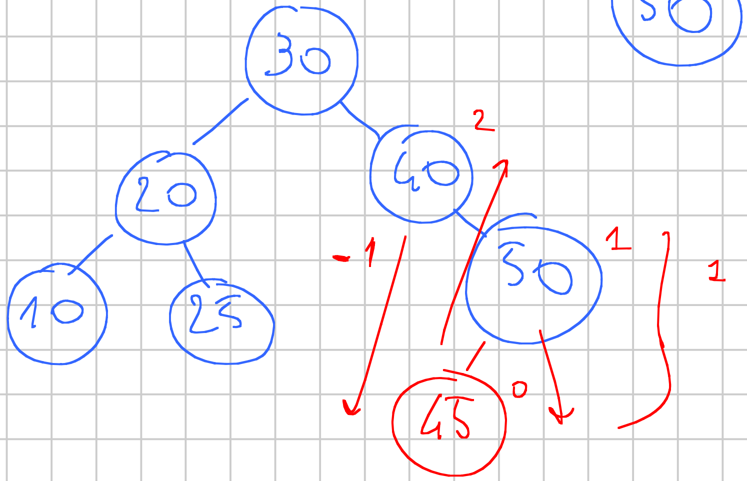
Insert(50)



20 è il nodo critico

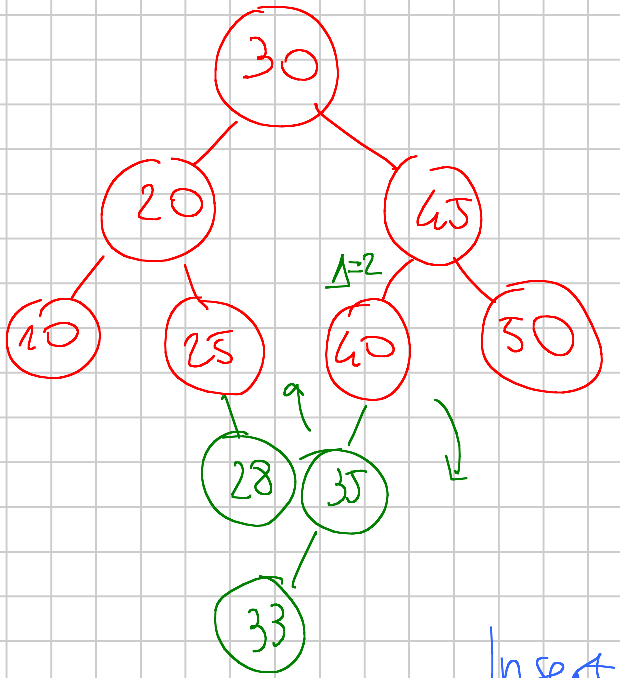
DD(20)

Insert(45)

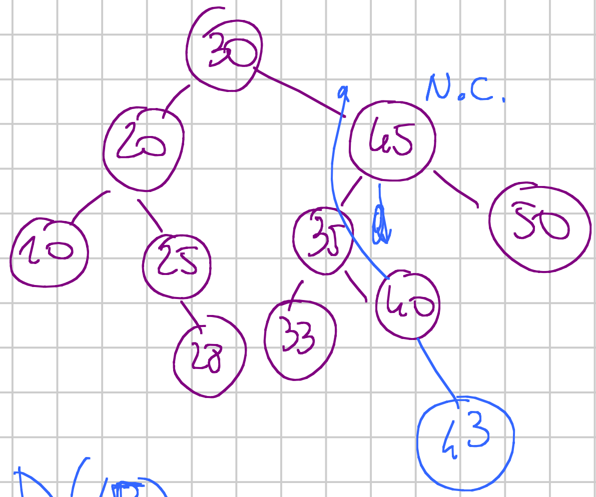


DS(40)

Insert (28, 35, 33)

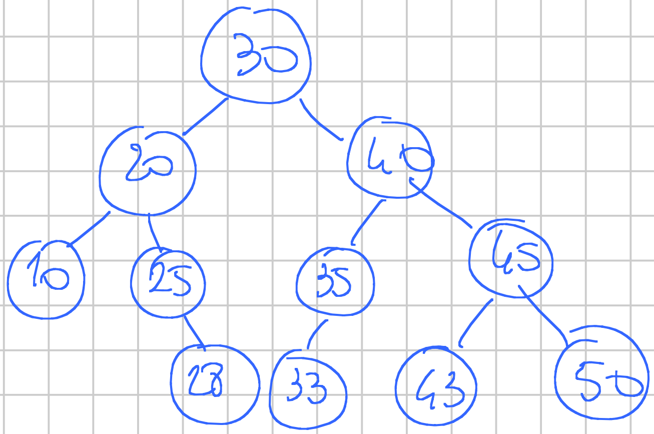


SS (40)



Insert(43)

SD(45)


30
20
40
10
25
35
45
28
33
43
50