VERIFICA DI UN ABR

Controlliamo per ogni nodo che il suo valore stia in un intervallo di valori (a,b).
Se la radice $x \in (a,b)$ allora il figlio sinistro deve stare in $(a, x-1)$ e il figlio destro $(x+1, b)$.
Essendo il controllo costante, a livello globale l'algoritmo sarà lineare su n $(n \cdot \Theta(1) = \Theta(n))$

```
CHECK ABR (T, a, b)
IF T ≠ NIL THEN
    IF a ≤ T->Key ≤ b THEN
        RET = CHECK ABR(T->sx, a, T->Key - 1)
        IF RET = TRUE            non ci sono violazioni
            RET = CHECK ABR(T->dx, T->Key + 1, b)
    ELSE
        RET = FALSE
ELSE
    RET = TRUE                  non sono state trovate violazioni fino alle foglie
RETURN RET
```

I valori di (a,b) all'inizio possono essere il minimo rappresentabile e il massimo valore rappresentabile
In qualche modo questa è una visita in order.

Abbiamo visto che in un ABR la ricerca dipende dalla struttura ($T_{SEARCH}(n) = O(h)$) e non dai nodi, in particolare $\Omega(\lg_2 n) = T_{SEARCH}(n) = O(n)$ (valori possibili per l'altezza).
Se riuscissimo a limitare l'altezza a $\lg_2 n$ avremmo
$T_{SEARCH}(n) = O(\lg_2 n) \Rightarrow \Theta(\lg_2 n)$
Vogliamo trovare un albero quindi che limiti l'altezza e mantenga gli inserimenti efficienti (risolvendo quindi il "problema" degli ABR).

ABR BILANCIATI

Gli ABR BILANCIATI sono ABR in cui $h(T) = O(\lg_2 n)$ per $|T| = n$ e sono chiusi rispetto alla ricerca e inserimento degli ABR (quindi mantengono le ottimizzazioni).

Definiamo T un ABR **PERFETTAMENTE BILANCIATO** se

$\forall x \in T \quad -1 \leq |T \to sx| - |T \to dx| \leq 1$ ovvero la quantità di nodi a destra e a sinistra è uguale al più di uno

Un ABR perfettamente bilanciato è bilanciato, **NON** vale il contrario

Purtroppo gli ABR (perfettamente) bilanciati non favoriscono un inserimento ottimizzato come quello degli ABR propio per le loro caratteristiche strutturali.

ALBERI AVL

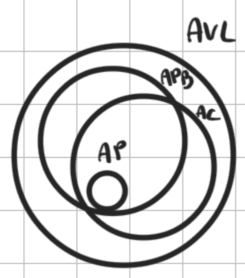
Sono ABR tali che $\forall x \in T \; -1 \leq h(T \to sx) - h(T \to dx) \leq 1$

Gli **AVL** hanno l'importante caratteristica di avere una forte correlazione tra altezza e numero di nodi (così come alberi pieni e alberi completi).

Inoltre soddisfano una proprietà fondamentale:

Sia T un AVL, $|T| = n \Rightarrow h(T) = O(\lg n)$

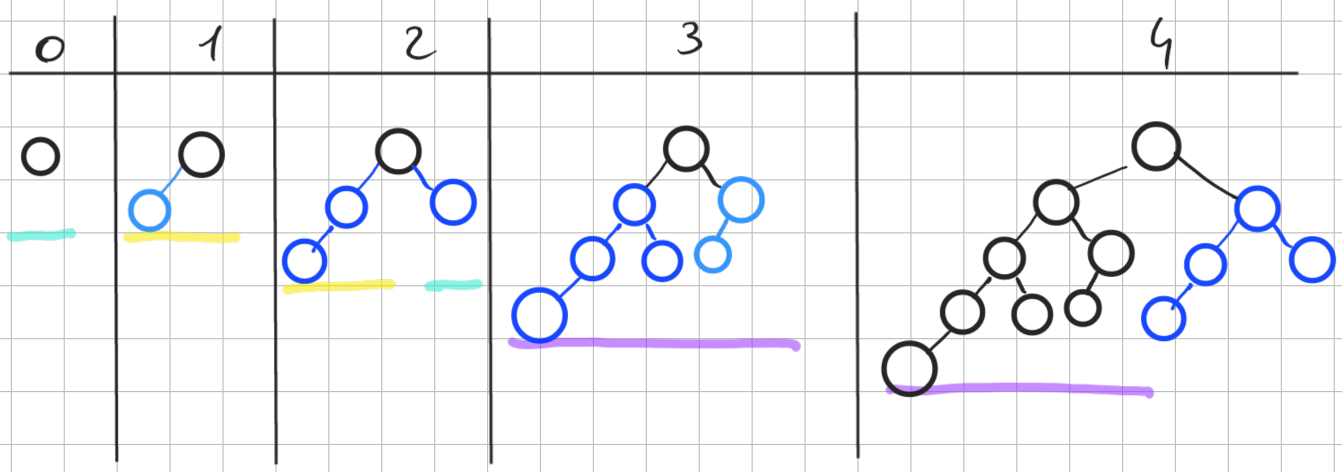
OSS



AVL MINIMI (di altezza h)

Fissata un altezza h definiamo AVL MINIMI di altezza h gli AVL di altezza h con il minimo numero di nodi.

Gli AVL minimi hanno per ogni nodo come sottoalbero sinistro un AVL minimo di altezza $h-1$ e come sottoalbero destro un AVL minimo di altezza $h-2$; inoltre sono tutti isomorfi tra di loro (fissato h) sul numero di nodi



PROPRIETA' DEGLI AVL MINIMI

Sia h l'altezza.

$\forall h \geq 0 \quad T_h \in AVL_M \quad h(T_h) = h \wedge |T_h| = n$

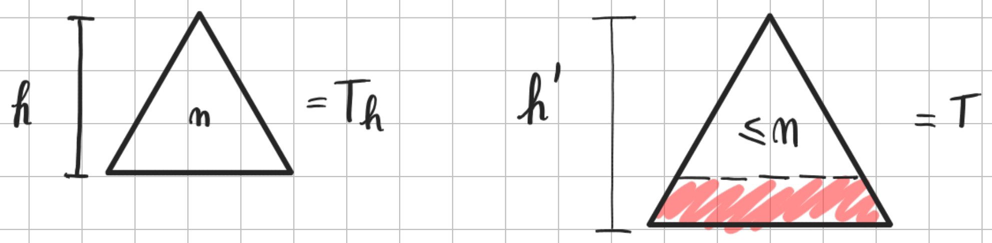
$\qquad \forall T \in AVL \quad |T| \leq n \quad h(T) \leq h$

DIMOSTRAZIONE

Per assurdo sia $h(T) > h$ e rinominiamo $h(t) = h'$

Se $h' > h$ significa che T ha dei nodi a livello più basso dell'albero T_h di altezza h.

Eliminiamo i nodi a livello più basso e otteniamo un nuovo



Chiamiamo il nuovo albero derivato da T, T' che avrà altezza h e numero di nodi $< n$ (perché abbiamo rimosso almeno un nodo).

Se T' è un AVL avremo un assurdo perché avremmo un AVL con stessa altezza ma numero inferiore di nodi, quindi T_h non sarebbe un AVL minimo.

Seppur la cancellazione non conserva per forza le proprietà degli AVL, T' e' sicuramente un AVL perché
abbiamo cancellato dal basso e questo tipo di cancellazione preserva le proprietà, vediamo perché:
Se per assurdo T' non e' un AVL significa che c'e' un nodo x tale che i suoi sottoalberi hanno differenza d'altezza maggiore di 1
Questa violazione pero' era per forza presente anche in T perché tutti i nodi di T' sono nodi di T, ma questo e' un assurdo perché T e' un AVL per ipotesi.

Quindi T' e' un AVL => T' e' un AVL minimo di altezza h

Assurdo! Perché T_h e' un AVL minimo di altezza h con un numero di nodi strettamente maggiore di T

OSS

Gli AVL minimi ci aiuteranno a dimostrare che l'altezza degli AVL e' limitata superiormente da $\lg_2 n$ perché fissato il numero di nodi hanno altezza massima, sono "i peggiori" sull'altezza, e se sono limitati superiormente da $\lg_2 n$ lo saranno sicuramente tutti gli AVL con lo stesso numero di nodi

NUMERO DI NODI DI UN AVL MINIMO

Definiamo una funzione che fissata l'altezza ci da il numero di nodi di un AVL minimo (abbiamo visto che e' unico) sapendo che per un generico albero T il numero di nodi e' $N(T) = 1 + N(T \to sx) + N(T \to dx)$

Ne deriva che per come l'abbiamo descritto, un AVL di altezza h ha numero di nodi

$N(h) = 1 + N(h-1) + N(h-2)$

Ricordando in particolare che $N(0) = 1$ e $N(1) = 2$ sono i casi base

Si puo' osservare un' analogia con la successione di Fibonacci che e' così definita

$F(0) = 0$, $F(1) = 1$

$F(x) = F(x-1) + F(x-2)$

Visualizziamo e confrontiamo $F(x)$ $F(h)$

	0	1	2	3	4	5	6	7	
F(x)	0	1	1	2	3	5	8	13	...
N(h)	1	2	4	7	12	20	33	...	

Dallo schema evince che $N(h) = F(h+3) - 1 \quad \forall h \geq 2$

Proviamo a dimostrarlo

DIMOSTRAZIONE

Dimostriamo per induzione su h con base 0

$h=0$ $\quad N(0)=1=F(3)-1$ ✓

esplicitiamo anche $N(1)$ che è

$h=1$ $\quad N(1)=2=F(4)-1$ ✓

$h\geq 2$ $\quad N(h)=1+N(h-1)+N(h-2)=\cancel{1}+F(h+2)-\cancel{1}+F(h-1)-1$

$=F(h+3)-1$ ✓

Fibonacci