Corso di laurea in Matematica
Insegnamento di Informatica generale
Lezioni in modalità mista o a distanza

Esercizi sulle equazioni di ricorrenza

Giancarlo Bongiovanni
Ivano Salvo

SAPIENZA
UNIVERSITÀ DI ROMA

Queste dispense sono state realizzate sulla base delle slide preparate da T. Calamoneri e G. Bongiovanni per il corso di Informatica Generale tenuto a distanza nell'A.A. 2019/20

Esercizio 1 – metodo iterativo

Risolveremo queste tre equazioni:

1.1 $T(n) = 2T(n/2) + \Theta(n)$

1.2 $T(n) = 3T(n/2) + \Theta(n)$

1.3 $T(n) = 2T(n/3) + \Theta(n)$

Per tutte:

$$T(1) = \Theta(1)$$

Sarà interessante vedere come cambiano i conteggi ed il risultato in funzione dei valori di a e b.

Esercizio 1.1 – metodo iterativo

Equazione da risolvere:

$T(n) = 2T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

Iniziamo con le sostituzioni:

$$
\begin{aligned}
T(n) \quad &= 2T(n/2) + \Theta(n) = \\
&= 2[2T(n/2^2) + \Theta(n/2^1)] + \Theta(n/2^0) = \\
&= 2[2[2T(n/2^3) + \Theta(n/2^2)] + \Theta(n/2^1)] + \Theta(n/2^0) = \\
&=2^3T(n/2^3) + 2^2\,\Theta(n/2^2) + 2^1\,\Theta(n/2^1) + 2^0\,\Theta(n/2^0) \\
&\;\dots \\
&=2^k\,T(n/2^k)+\sum_{i=0}^{k-1} 2^i \Theta\left(\frac{n}{2^i}\right)
\end{aligned}
$$

Segue esercizio 1.1

Come abbiamo già visto, proseguiamo finché

$$2^k = n, \text{ ossia } k = \log n$$

Ottenendo

$$T(n) = 2^{\log n}\,\Theta(1) + \sum_{i=0}^{\log n-1} 2^i\Theta\left(\frac{n}{2^i}\right) =$$

$$= 2^{\log n}\,\Theta(1) + n\sum_{i=0}^{\log n-1} 2^i\Theta\left(\frac{1}{2^i}\right) =$$

$$= 2^{\log n}\,\Theta(1) + n\sum_{i=0}^{\log n-1} \Theta\left(\frac{2^i}{2^i}\right) =$$

$$= 2^{\log n}\,\Theta(1) + n\sum_{i=0}^{\log n-1} \Theta(1)$$

Segue esercizio 1.1

Ricordiamo che

$$2^{\log n} = n = \Theta(n)$$

e che, banalmente,

$$\sum_{i=0}^{\log n - 1} \Theta(1) = \Theta(\log n)$$

quindi

$$n \sum_{i=0}^{\log n - 1} \Theta(1) = \Theta(n \log n)$$

Da cui otteniamo

$$T(n) = \Theta(n) + \Theta(n \log n) = \Theta(n \log n)$$

Esercizio 1.2 – metodo iterativo

Equazione da risolvere:

$T(n) = 3T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

Iniziamo con le sostituzioni:

$$
\begin{aligned}
T(n) &= 3T(n/2) + \Theta(n) = \\
&= 3[3T(n/2^2) + \Theta(n/2^1)] + \Theta(n/2^0) = \\
&= 3[3[3T(n/2^3) + \Theta(n/2^2)] + \Theta(n/2^1)] + \Theta(n/2^0) = \\
&= 3^3 T(n/2^3) + 3^2\,\Theta(n/2^2) + 3^1\,\Theta(n/2^1)) + 3^0\,\Theta(n/2^0) \\
&\ \dots \\
&= 3^k\,T(n/2^k) + \sum_{i=0}^{k-1} 3^i \Theta\left(\frac{n}{2^i}\right)
\end{aligned}
$$

Segue esercizio 1.2

Come abbiamo già visto, proseguiamo finché

$$2^k = n, \text{ ossia } k = \log n$$

Ottenendo

$$T(n) = 3^{\log n}\,\Theta(1) + \sum_{i=0}^{\log n-1} 3^i \Theta\left(\frac{n}{2^i}\right) =$$

$$= 3^{\log n}\,\Theta(1) + n\sum_{i=0}^{\log n-1} 3^i \Theta\left(\frac{1}{2^i}\right) =$$

$$= 3^{\log n}\,\Theta(1) + n\sum_{i=0}^{\log n-1} \Theta\left(\left(\frac{3}{2}\right)^i\right)$$

Segue esercizio 1.2

Occupiamoci del primo termine della somma, cambiando base al logaritmo. Ricordiamo che

$$\log_x n = \log_x y \log_y n$$

E dunque

$$3^{\log_2 n} = 3^{\log_2 3 \log_3 n} = n^{\log_2 3} = \Theta(n^{\log_2 3})$$

NOTA: quando all'equazione di ricorrenza si può applicare il teorema principale, il primo termine della somma risulta sempre essere $n^{log_b a}$ (in questo esempio a=3 e b=2) e questo spiega perché nel teorema principale esso è uno dei due elementi che si confrontano.

Segue esercizio 1.2

Valutiamo ora la sommatoria presente nel secondo termine della somma:

$$\sum_{i=0}^{\log n-1} \left(\frac{3}{2}\right)^i$$

Sappiamo che essa vale

$$\sum_{i=0}^{\log n-1} \left(\frac{3}{2}\right)^i = \frac{\left(\frac{3}{2}\right)^{\log n} - 1}{\frac{3}{2} - 1} = 2\left(\left(\frac{3}{2}\right)^{\log n} - 1\right)$$

Ma

$$\left(\frac{3}{2}\right)^{\log n} = \frac{3^{\log n}}{2^{\log n}} = \frac{3^{\log_2 3 \log_3 n}}{n} = \frac{n^{\log_2 3}}{n}$$

e quindi

$$n \sum_{i=0}^{\log n-1} \Theta\left(\left(\frac{3}{2}\right)^i\right) = n\ \Theta\left(\frac{n^{\log_2 3}}{n}\right) = \Theta(n^{\log_2 3})$$

Segue esercizio 1.2

Ricapitolando:

$$3^{\log n}\,\Theta(1) = \Theta(n^{\log_2 3})$$

$$\sum_{i=0}^{\log n-1} 3^i \Theta\left(\frac{n}{2^i}\right) = \Theta(n^{\log_2 3})$$

E quindi per l'equazione:

$T(n) = 3T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

Abbiamo:

$$T(n) = 3^{\log n}\,\Theta(1) + \sum_{i=0}^{\log n-1} 3^i \Theta\left(\frac{n}{2^i}\right) =$$

$$\mathbf{= \Theta(n^{log_2 3}) + \Theta(n^{log_2 3}) = \Theta(n^{log_2 3})}$$

Esercizio 1.3 – metodo iterativo

Equazione da risolvere:

$T(n) = 2T(n/3) + \Theta(n)$

$T(1) = \Theta(1)$

Iniziamo con le sostituzioni:

$$
\begin{aligned}
T(n) &= 2T(n/3) + \Theta(n) = \\
&= 2[2T(n/3^2) + \Theta(n/3^1)] + \Theta(n/3^0) = \\
&= 2[2[2T(n/3^3) + \Theta(n/3^2)] + \Theta(n/3^1)] + \Theta(n/3^0) = \\
&=2^3 T(n/3^3) + 2^2\, \Theta(n/3^2) + 2^1\, \Theta(n/3^1) + 2^0\, \Theta(n/3^0) \\
&\ \dots \\
&=2^k\, T(n/3^k)+\sum_{i=0}^{k-1} 2^i \Theta\left(\frac{n}{3^i}\right)
\end{aligned}
$$

Segue esercizio 1.3

Proseguiamo finché

$$3^k = n, \text{ ossia } k = \log_3 n$$

Ottenendo

$$T(n) = 2^{\log_3 n}\Theta(1) + \sum_{i=0}^{\log_3 n-1} 2^i \Theta\left(\frac{n}{3^i}\right) =$$

$$= 2^{\log_3 2 \log_2 n}\,\Theta(1) + n\sum_{i=0}^{\log_3 n-1} 2^i \Theta\left(\frac{1}{3^i}\right) =$$

$$= n^{\log_3 2}\,\Theta(1) + n\sum_{i=0}^{\log_3 n-1} \Theta\left(\left(\frac{2}{3}\right)^i\right) =$$

$$= \Theta(n^{\log_3 2}) + n\sum_{i=0}^{\log_3 n-1} \Theta\left(\left(\frac{2}{3}\right)^i\right)$$

Segue esercizio 1.3

Valutiamo ora la sommatoria presente nel secondo termine della somma:

$$\sum_{i=0}^{\log n-1} \left(\frac{2}{3}\right)^i$$

Sappiamo che per essa vale:

$$1 = \sum_{i=0}^{0} \left(\frac{2}{3}\right)^i \leq \sum_{i=0}^{\log n-1} \left(\frac{2}{3}\right)^i \leq \sum_{i=0}^{\infty} \left(\frac{2}{3}\right)^i = \frac{1}{1-\frac{2}{3}} = 3$$

e quindi

$$n \sum_{i=0}^{\log n-1} \left(\frac{2}{3}\right)^i = n\,\Theta(1) = \Theta(n)$$

Segue esercizio 1.3

Ricapitolando, per l'equazione:

$$T(n) = 2T(n/3) + \Theta(n)$$
$$T(1) = \Theta(1)$$

abbiamo:

$$T(n) = \Theta(n^{\log_3 2}) + n \sum_{i=0}^{\log_3 n-1} \Theta\left(\left(\frac{2}{3}\right)^i\right) =$$

$$\boldsymbol{= \Theta(n^{log_3 2}) + \Theta(n) = \Theta(n)}$$

Reminder

C'è una sommatoria che capita spesso di dover risolvere:

$$\sum_{i=0}^{k} x^i$$

Se 0 < x < 1 sappiamo che:

$$\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$$

E quindi la sommatoria $\sum_{i=0}^{\infty} x^i$ *è Θ(1) dato che*

$$1 \leq \sum_{i=0}^{k} x^i \leq \frac{1}{1-x}$$

Se invece x >1 si può usare la soluzione:

$$\sum_{i=0}^{k} x^i = \frac{x^{k+1}-1}{x-1}$$

Esercizio 2 – metodo principale

Risolveremo queste tre equazioni:

2.1 $T(n) = 2T(n/2) + \Theta(n)$

2.2 $T(n) = 3T(n/2) + \Theta(n)$

2.3 $T(n) = 2T(n/3) + \Theta(n)$

Per tutte:

$$T(1) = \Theta(1)$$

Sarà interessante vedere come cambia il caso del teorema generale per le tre equazioni.

Esercizio 2.1 – metodo principale

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n)$$
$$T(1) = \Theta(1)$$

- $a = 2,\ b = 2$
- $f(n) = \Theta(n)$
- $n^{\log_b a} = n^{\log_2 2} = n$

Poiché

$$f(n) = \Theta(n^{\log_b a})$$

siamo nel **caso 2**, per cui

$$\boldsymbol{T(n) = \Theta(n \log n)}$$

Esercizio 2.2 – metodo principale

Equazione da risolvere:

$T(n) = 3T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

- $a = 3, b = 2$
- $f(n) = \Theta(n)$
- $n^{\log_b a} = n^{\log_2 3} = n^{1+\varepsilon}$ con $\varepsilon > 0$

Poiché

$$f(n) = \Theta(n^{\log_b a - \varepsilon})$$

siamo nel **caso 1**, per cui

$$\boldsymbol{T(n) = \Theta(n^{\log_b a}) = \Theta(n^{\log_2 3})}$$

Esercizio 2.3 – metodo principale

Equazione da risolvere:

T(n) = 2T(n/3) + Θ(n)

T(1) = Θ(1)

- *a = 2, b = 3*
- *f(n) = Θ(n)*
- $n^{\log_b a} = n^{\log_3 2} = n^{1-\varepsilon}$ con ε > 0

Poiché

$$f(n) = \Theta(n^{\log_b a+\varepsilon})$$

siamo nel **caso 3**, per cui

$$\boldsymbol{T(n) = f(n) = \Theta(n)}$$

se riusciamo a dimostrare che $af(\frac{n}{b}) \leq c\, f(n)$ per qualche c <1.

Ponendo c=2/3 si ha:

$$2\frac{n}{3} \leq \frac{2}{3}\, n$$

Esercizio 3 – equazione risolta con quattro metodi

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n)$$

$$T(1) = \Theta(1)$$

Metodo iterativo: già fatto

Metodo principale: già fatto

Metodo di sostituzione: esercizio 3.1

Metodo dell'albero: esercizio 3.2

Esercizio 3.1 – metodo di sostituzione

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n)$$

$$T(1) = \Theta(1)$$

Dobbiamo innanzi tutto eliminare la notazione asintotica, quindi l'equazione diventa:

$T(n) = 2T(n/2) + cn$ per qualche costante $c > 0$

$T(1) = d$ per qualche costante $d > 0$

Segue esercizio 3.1

Proviamo a dimostrare per induzione che la soluzione è:

$$T(n) \geq kn \log n$$

dove k è una costante da determinare.

Passo base

$T(1) = d \geq k1 \log 1 = 0$, che è sempre verificata.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$$T(n) \geq 2\,(k\, n/2 \log n/2) + cn = kn \log n/2 + cn =$$

$$= kn\,(\log n - 1) + cn = kn \log n - kn + cn \geq kn \log n$$

se e solo se $cn \geq kn$ cioè se e solo se $c \geq k$. Poiché un tale k è sempre possibile da trovare, ne concludiamo che

$$T(n) = \Omega(n \log n).$$

Segue esercizio 3.1

Per quanto riguarda la maggiorazione, non possiamo usare

$$T(n) \leq k' \, n \log n$$

perché in tal modo il passo base non è verificato, infatti non esiste una costante k' positiva tale che

$$T(1) = d \leq k' \, 1 \log 1 = 0$$

Tentiamo allora con

$$T(n) \leq k' \, n \log n + h.$$

Segue esercizio 3.1

Tentiamo

$$T(n) \leq k' n \log n + h.$$

Passo base

$T(1) = d \leq h$ che è vera per opportuni valori di h.

Passo induttivo

Sostituendo l'ipotesi induttiva nell'equazione otteniamo:

$T(n) \leq 2\,(k' n/2 \log n/2 + h) + cn =$

$= k' n \log n/2 + 2h + cn =$

$= k'n (\log n - 1) + 2h + cn =$

$= k'n \log n + h - k'n + cn + h$

$\leq k'n \log n + h$

se e solo se $-k'n + cn + h \leq 0$, ossia $cn + h \leq k'n$

Segue esercizio 3.1

Anche in questo caso esistono opportuni valori di h e k', fissato c, per cui la disuguaglianza è verificata. Ne segue che $T(n)=O(n \log n)$.

Avendo precedentemente dimostrato che $T(n)=\Omega(n \log n)$ *e* mettendo insieme le due notazioni asintotiche trovate, si deduce che l'equazione di ricorrenza

$$T(n) = 2T(n/2) + \Theta(n)$$
$$T(1) = \Theta(1)$$

ha soluzione

$$\boldsymbol{T(n)= \Theta(n \log n)}$$

Esercizio 3.2 – metodo dell'albero

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n)$$

$$T(1) = \Theta(1)$$

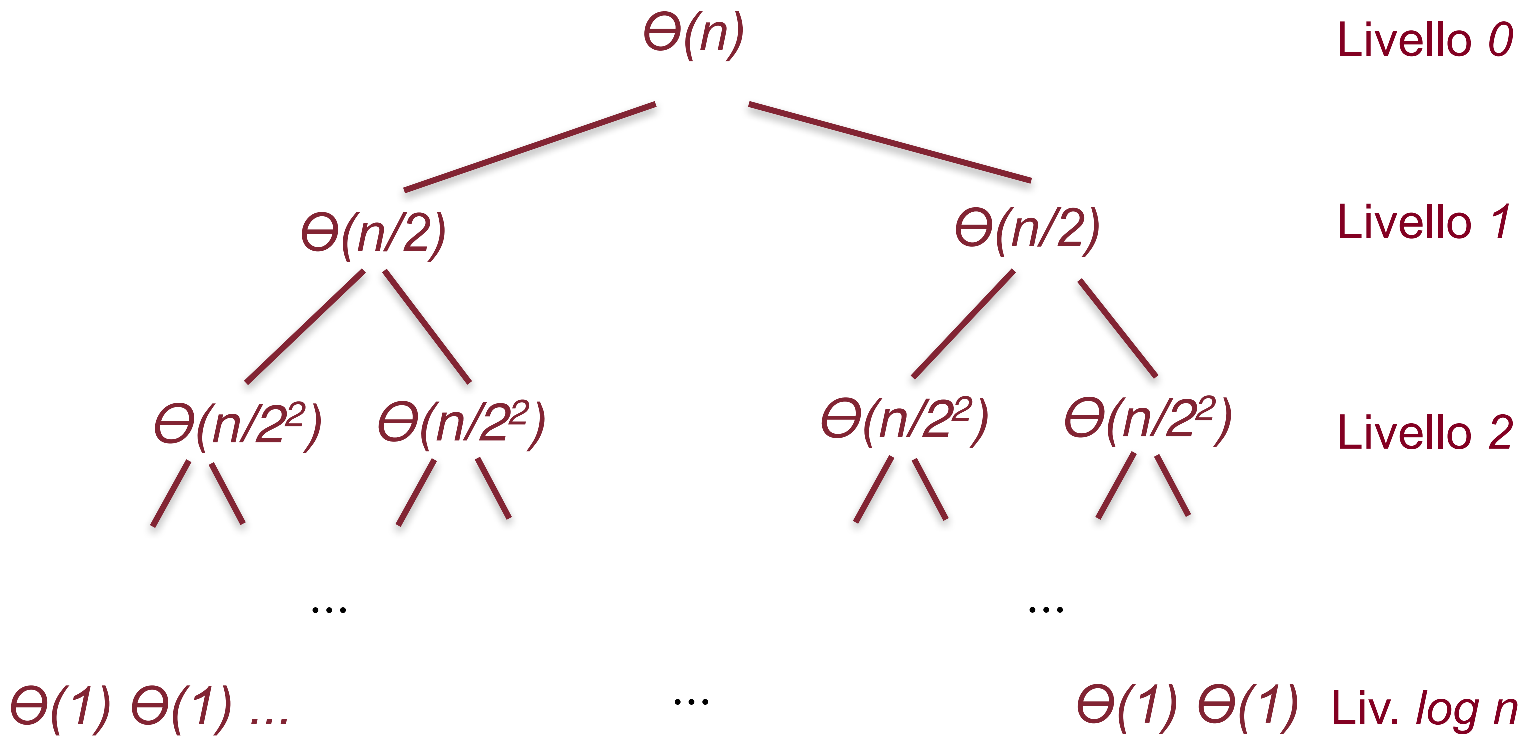
La radice dell'albero dà un contributo di $\Theta(n)$ ed ha due figli, entrambi etichettati con $T(n/2)$:

$\Theta(n)$

$T(n/2)$ $T(n/2)$

Segue esercizio 3.2

Iterando il procedimento otteniamo questo albero, nel quale numeriamo i livelli da zero (radice) a *log n*:



Segue esercizio 3.2

Il contributo della generica riga *i*-esima dell'albero è dato dal valore su ciascuno dei suoi nodi, pari a $\Theta(n/2^i)$, moltiplicato per il numero di nodi, cioè 2^i.

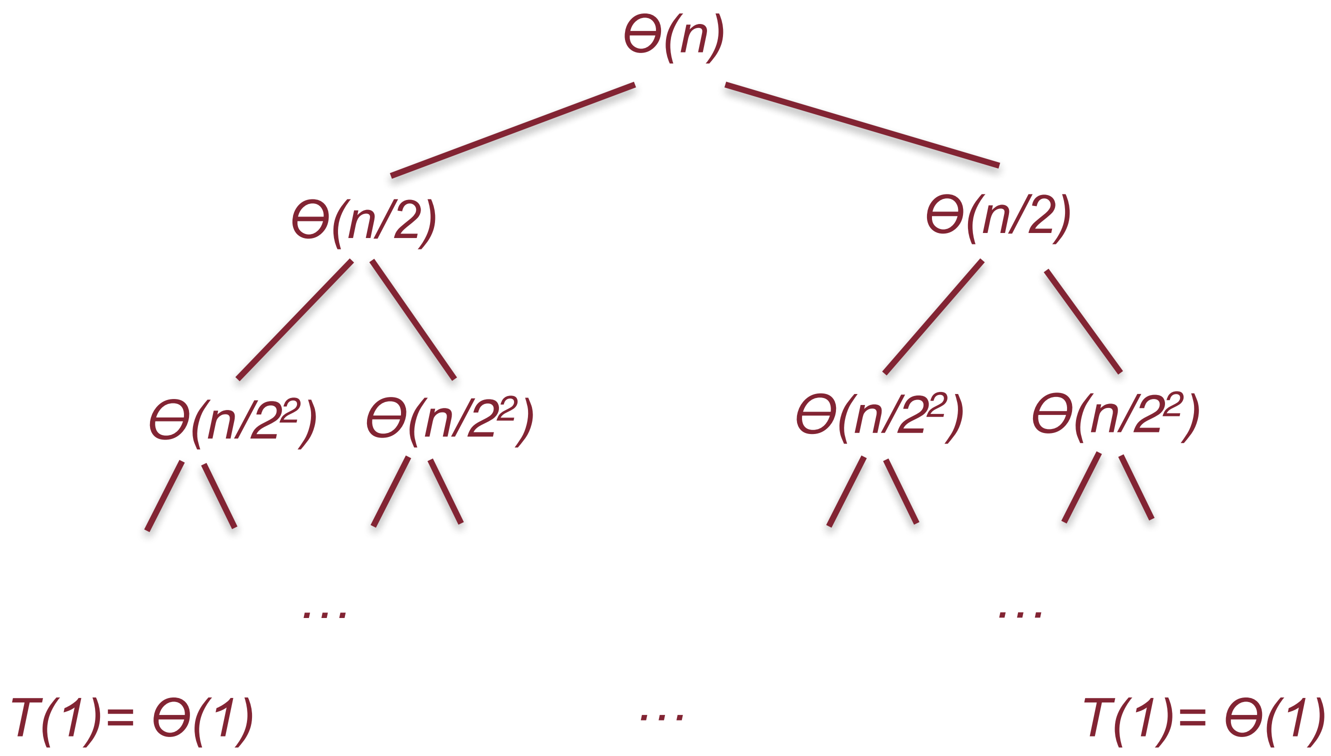
Considerato che le righe sono *log n* + 1 (da 0 a *log n*), si ha:

$$T(n) = \sum_{i=0}^{\log n} \Theta\left(\frac{n}{2^i}\right) 2^i = \sum_{i=0}^{\log n} \Theta(n) =$$

$$= \Theta(n\,(\log n + 1)) =$$
$$= \Theta(n \log n) + \Theta(n) =$$
$$= \boldsymbol{\Theta(n \log n)}$$

Confronto fra conteggi

Metodo dell'albero



Metodo Principale

$$\sum_{i=0}^{\log n-1} 2^i \Theta\left(\frac{n}{2^i}\right)$$

$$2^{\log n}\,\Theta(1)$$

Esercizio 4 – equazione risolta con quattro metodi

Equazione da risolvere:

$$T(n) = 4T(n/2) + \Theta(n)$$

$$T(1) = \Theta(1)$$

Metodo iterativo: esercizio 4.1

Metodo principale: esercizio 4.2

Metodo di sostituzione: esercizio 4.3

Metodo dell'albero: esercizio 4.4

Esercizio 4.1 – metodo iterativo

Equazione da risolvere:

$T(n) = 4T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

$T(n)=4T(n/2) + \Theta(n) = 4(4T(n/2^2) + \Theta(n/2^1)) + \Theta(n) =$

$=4^2T(n/2^2) + 4^1\Theta(n/2^1) + \Theta(n/2^0) =$

$=4^2(4T(n/2^3) + \Theta(n/2^2)) +4^1\Theta(n/2^1) + \Theta(n/2^0) =$

$=4^3T(n/2^3) + 4^2\Theta(n/2^2) + 4^1\Theta(n/2^1) + 4^0\Theta(n/2^0) =\ldots=$

$= 4^k\ T(n/2^k) + \sum_{i=0}^{k-1} 4^i \Theta\left(\frac{n}{2^i}\right) = 4^k\ T(n/2^k) + \sum_{i=0}^{k-1} 2^i \Theta(n) =$

$= 4^k\ T(n/2^k) + \Theta(n) \sum_{i=0}^{k-1} 2^i$

Segue esercizio 4.1

Equazione da risolvere:

$T(n) = 4T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

Ci fermiamo quando $n/2^k=1$ *cioè* $k=\log n$, *ottenendo*

$T(n) = 4^{\log n}\, T(1) + \Theta(n) \sum_{i=0}^{\log n-1} 2^i$

Ricordando che $4^{\log n} = n^2$ *e che* $\sum_{i=0}^{\log n-1} 2^i = 2^{\log n} - 1 = \Theta(n))$

Otteniamo

$$T(n) = n^2\, \Theta(1) + \Theta(n)\, \Theta(n) = \Theta(n^2)$$

Dunque la soluzione tramite questo metodo è $\boldsymbol{\Theta(n^2)}$

Esercizio 4.2 – metodo principale

Equazione da risolvere:

$T(n) = 4T(n/2) + \Theta(n)$

$T(1) = \Theta(1)$

L'equazione di ricorrenza soddisfa le ipotesi del teorema.

- $a = 4,\ b = 2$
- $n^{log_b a} = n^{log_2 4} = n^2$
- $f(n) = \Theta(n) = \Theta(n^1) = O\left(n^{\log_b a - \varepsilon}\right)$ (ε=1)

Siamo quindi nel **caso 1**, per cui:

$$\boldsymbol{T(n) = \Theta(n^2)}$$

Esercizio 4.3 – metodo di sostituzione

Dobbiamo innanzi tutto eliminare la notazione asintotica, quindi l'equazione diventa:

$T(n) = 4T(n/2) + cn$ per qualche costante c > 0

$T(1) = d$ per qualche costante d > 0

Proviamo a dimostrare per induzione che la soluzione è:

$\boldsymbol{T(n) \geq kn^2}$, dove k è una costante da determinare.

Passo base

$T(1) = d \geq k$, da cui deduciamo una prima condizione su k.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$T(n) \geq 4\,k\,(n/2)^2 + cn = 4kn^2/4 + cn = kn^2 + cn \geq kn^2$ sempre.

Ne concludiamo che

$$\boldsymbol{T(n) = \Omega(n^2)}$$

Segue esercizio 4.3

$T(n) = 4T(n/2) + cn$

$T(1) = d$

Per quanto riguarda la maggiorazione, tentiamo la soluzione

$$\boldsymbol{T(n) \leq k' n^2}$$

dove k' è una costante da determinare.

Passo base

$T(1) = d \leq k'$, da cui deduciamo una prima condizione su k'.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$$T(n) \leq 4\, k' (n/2)^2 + cn = k' n^2 + cn$$

che non si può dimostrare sia $\leq kn^2$ perché c è una costante positiva.

Segue esercizio 4.3

Tentiamo allora

$$T(n) \leq k' n^2 - hn.$$

Passo base

$T(1) = d \leq k' - h$ che è vera per certi valori di k' e h.

Passo induttivo

Sostituendo l'ipotesi induttiva nell'equazione otteniamo:

$$T(n) \leq 4\,(k'(n/2)^2 - hn) + cn = k'n^2 - 4hn + cn = k'n^2 - hn - 3hn + cn$$

Questa quantità è $\leq k'n^2 - hn$ se e solo se $-3hn + cn \leq 0$, cioè se e solo se $h \geq c/3$.

Ne concludiamo che $T(n) = O(n^2)$ e, unendo i due risultati:

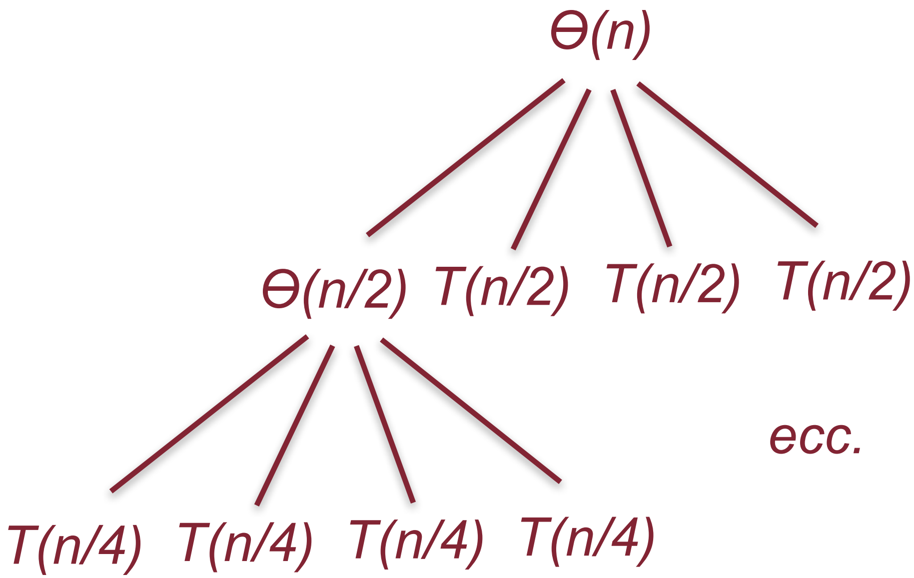
$$T(n) = \Theta(n^2)$$

Esercizio 4.4 – metodo dell'albero

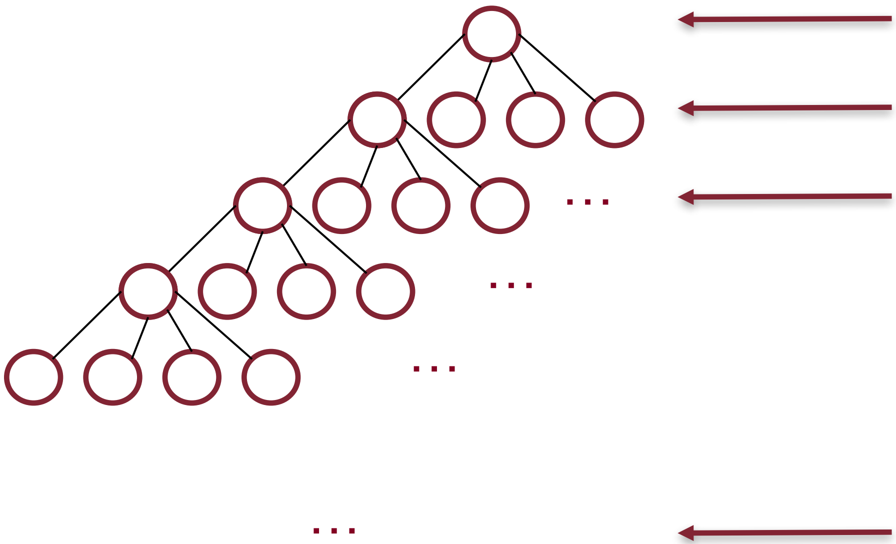
Equazione da risolvere:

$$T(n) = 4T(n/2) + \Theta(n)$$
$$T(1) = \Theta(1)$$

Il nodo T(n) «genera» un costo di Θ(n) più quattro figli, ciascuno relativo a un sottoproblema di dimensione n/2:



Segue esercizio 4.4



#nodi	contributo nodo
$1 = 4^0$	$\Theta(n)$
$4 = 4^1$	$\Theta(n/2)$
$16 = 4^2$	$\Theta(n/4)$
…	…
4^i	$\Theta(n/2^i)$

Sommando tutti i contributi:

$$T(n) = \sum_{i=0}^{\log n} 4^i \Theta\left(\frac{n}{2^i}\right) = \Theta(n) \sum_{i=0}^{\log n} 2^i = \Theta(n)\left(2^{\log n+1} - 1\right) = \Theta(n)(2n-1) = \Theta(n^2)$$

Esercizio 5 – equazione risolta con quattro metodi

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n^2)$$

$$T(1) = \Theta(1)$$

Metodo iterativo: esercizio 5.1

Metodo principale: esercizio 5.2

Metodo di sostituzione: esercizio 5.3

Metodo dell'albero: esercizio 5.4 (lo rivediamo)

Esercizio 5.1 – metodo iterativo

Equazione da risolvere:

$T(n) = 2T(n/2) + \Theta(n^2)$

$T(1) = \Theta(1)$

$T(n) = 2T(n/2) + \Theta(n^2)$

$= 2(2T(n/2^2) + \Theta(\left(\frac{n}{2}\right)^2) + \Theta(n^2)$

$= 2(2(2T(n/2^3) + \Theta(\left(\frac{n}{4}\right)^2) + \Theta(\left(\frac{n}{2}\right)^2)) + \Theta(n^2) =$

$\ldots$

$= 2^k T(n/2^k) + \sum_{i=0}^{k-1} 2^i \Theta\left(\left(\frac{n}{2^i}\right)^2\right) = 2^k T(n/2^k) + \sum_{i=0}^{k-1} \frac{1}{2^i} \Theta(n^2) =$

$= 2^k T(n/2^k) + \Theta(n^2) \sum_{i=0}^{k-1} \left(\frac{1}{2}\right)^i$

Segue esercizio 5.1

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n^2)$$

$$T(1) = \Theta(1)$$

Ci fermiamo quando $n/2^k=1$ *cioè* $k=\log n$*, ottenendo*

$$T(n) = 2^{\log n}\, T(1) + \Theta(n^2) \sum_{i=0}^{\log n-1} \left(\frac{1}{2}\right)^i$$

Ricordando che $2^{\log n} = n$ *e che* $\sum_{i=0}^{\log n-1} \left(\frac{1}{2}\right)^i = \Theta(1)$

Otteniamo

$$T(n) = \Theta(n)+ \Theta(n^2)= \Theta(n^2)$$

Dunque la soluzione tramite questo metodo è $\boldsymbol{\Theta(n^2)}$

Esercizio 5.2 – metodo principale

Equazione da risolvere:

$T(n) = 2T(n/2) + \Theta(n^2)$

$T(1) = \Theta(1)$

- $a = 2,\ b = 2$
- $n^{log_b a} = n^{log_2 2} = n$
- $f(n) = \Theta(n^2) = \Theta(n^2) = \Omega\left(n^{\log_b a + \varepsilon}\right)$ (ε=1)

Siamo quindi nel **caso 3**, per cui:

$$\boldsymbol{T(n) = \Theta(n^2)}$$

perché ponendo c = ½ si ha:

$$a\left(\frac{n}{b}\right)^2 = 2\left(\frac{n}{2}\right)^2 \leq cn^2 = \frac{1}{2}n^2$$

Esercizio 5.3 – metodo di sostituzione

Dobbiamo innanzi tutto eliminare la notazione asintotica, quindi l'equazione diventa:

$T(n) = 2T(n/2) + cn^2$ per qualche costante c > 0

$T(1) = d$ per qualche costante d > 0

Proviamo a dimostrare per induzione che la soluzione è:

$\boldsymbol{T(n) \leq kn^2}$, dove k è una costante da determinare.

Passo base

$T(1) = d \leq k$, da cui deduciamo una prima condizione su k.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$T(n) \leq 2\,k\,(n/2)^2 + cn^2 = kn^2/2 + cn^2 = (k/2+c)n^2 \leq kn^2$ se $c \leq k/2$.

Ne concludiamo che

$$\boldsymbol{T(n)= O(n^2)}$$

Segue esercizio 5.3

Proviamo ora a dimostrare per induzione che la soluzione è:

$T(n) \geq hn^2$, dove h è una costante da determinare.

Passo base

$T(1) = d \geq h$, da cui deduciamo una prima condizione su h.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$T(n) \geq 2\, h\, (n/2)^2 + cn^2 = hn^2/2 + cn^2 = (h/2+c)n^2 \geq hn^2$

se $(h/2+c) \geq h$, ossia $h \leq 2c$.

Ne concludiamo che

$$T(n)= \Omega(n^2)$$

E quindi

$$T(n)= \Theta(n^2)$$

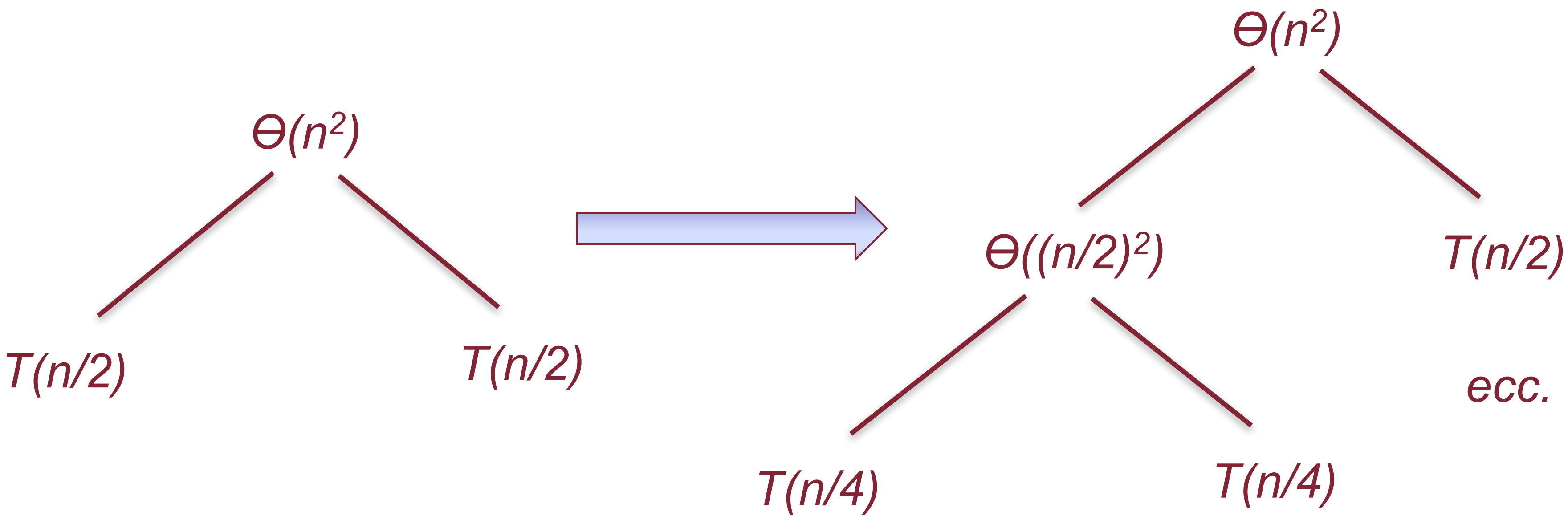
Esercizio 5.4 – metodo dell'albero

Equazione da risolvere:

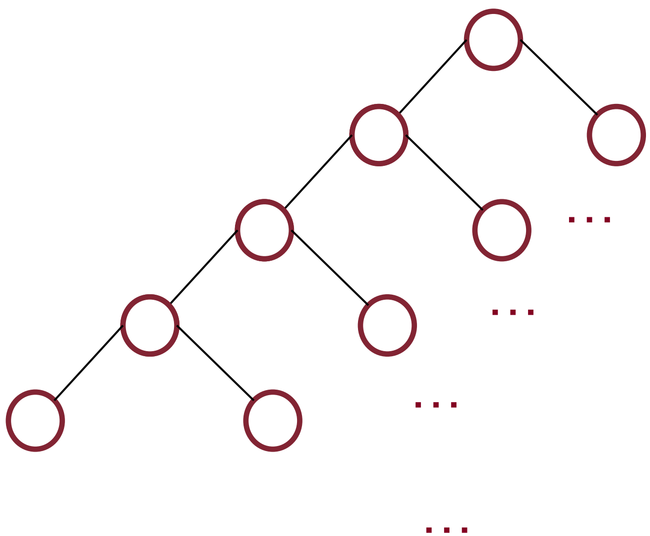
$$T(n) = 2T(n/2) + \Theta(n^2)$$

$$T(1) = \Theta(1)$$

Il nodo T(n) «genera» un costo di $\Theta(n^2)$ più due figli, ciascuno relativo a un sottoproblema di dimensione n/2:



Segue esercizio 5.4



	#nodi	contributo nodo
←	$1 = 2^0$	$\Theta(n^2)$
←	$2 = 2^1$	$\Theta\left(\left(\frac{n}{2}\right)^2\right)$
←	$2 = 2^2$	$\Theta\left(\left(\frac{n}{4}\right)^2\right)$
	…	…
←	2^i	$\Theta\left(\left(\frac{n}{2^i}\right)^2\right)$

Sommando tutti i contributi:

$$T(n) = \sum_{i=0}^{\log n} 2^i \Theta\left(\left(\frac{n}{2^i}\right)^2\right) = \Theta(n^2) \sum_{i=0}^{\log n} \left(\frac{1}{2}\right)^i = \Theta(n^2)$$

Esercizio 6 – equazione risolta con due metodi

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n \log n)$$

$$T(1) = \Theta(1)$$

Come abbiamo già visto essa non può essere risolta col metodo principale. Risolviamola con altri due metodi.

Metodo iterativo: esercizio 6.1

Metodo di sostituzione: esercizio 6.2

Metodo dell'albero: lasciato come esercizio

Esercizio 6.1 – metodo iterativo

Equazione da risolvere:

$T(n) = 2T(n/2) + \Theta(n \log n)$

$T(1) = \Theta(1)$

$T(n) = 2T(n/2) + \Theta(n \log n)$

$= 2(2T(\frac{n}{4}) + \Theta(\frac{n}{2}\log\frac{n}{2})) + \Theta(n \log n) =$

$= 2(2(2T(n/2^3) + \Theta(\frac{n}{4}\log\frac{n}{4})) + \Theta(\frac{n}{2}\log\frac{n}{2})) + \Theta(n \log n) =$

$\ldots$

$= 2^k T(\frac{n}{2^k}) + \sum_{i=0}^{k-1} \Theta\left(n \log \frac{n}{2^i}\right)$

Segue esercizio 6.1

Fermandoci al valore $k = \log n$ otteniamo:

$$T(n) = 2^{\log n} T(1) + \Theta\left(\sum_{i=0}^{\log n - 1} n \log \frac{n}{2^i}\right) =$$

$$= n\Theta(1) + \Theta\left(n \sum_{i=0}^{\log n - 1} \log n \; - \; n \sum_{i=0}^{\log n - 1} \log 2^i\right) =$$

$$= \Theta(n) + \Theta(n \log^2 n - n \sum_{i=0}^{\log n - 1} i \,) =$$

$$= \Theta(n) + \Theta(n \log^2 n - n \frac{\log n(\log n - 1)}{2}) =$$

$$= \Theta(n) + \Theta(n \log^2 n - \frac{n}{2} \log^2 n + \frac{n}{2} \log n) =$$

$$= \boldsymbol{\Theta(n \log^2 n)}$$

Esercizio 6.2 – metodo di sostituzione

Equazione da risolvere:

$$T(n) = 2T(n/2) + \Theta(n \log n)$$

$$T(2) = \Theta(1)$$

Impostiamo la dimensione del caso base a 2, per evitare di dover gestire il caso di log 1 = 0.

Dobbiamo innanzi tutto eliminare la notazione asintotica, quindi l'equazione diventa:

$T(n) = 2T(n/2) + an \log n$ per qualche costante $a > 0$

$T(2) = b$ per qualche costante $b > 0$

Segue esercizio 6.2

Proviamo a dimostrare per induzione che la soluzione è:

$T(n) \leq cn\, log^2 n$, dove *c* è una costante da determinare.

Passo base

*T(2) = b ≤ c*2*1*, vera per $c \geq b/2$.

Passo induttivo

Sostituendo nell'equazione generica otteniamo:

$T(n) \leq 2\, c\, \frac{n}{2}\, log^2 \frac{n}{2} + an\, log\, n =$

$= cn(log\, n - 1)^2 + an\, log\, n =$

$= cn(log^2 n - 2\, log\, n + 1) + an\, log\, n =$

$= cn\, log^2 n - 2\, cn\, log\, n + cn + an\, log\, n$

Segue esercizio 6.2

$$T(n) \le 2\,c\,\frac{n}{2}\,log^2\frac{n}{2} + an\log n =$$

$$= cn\,(log\,n - 1)^2 + an\log n =$$

$$= cn(log^2 n - 2\log n + 1) + an\log n =$$

$$= cn\,log^2 n - 2\,cn\log n + cn + an\log n$$

Ora, $cn\log n \ge cn$, quindi eliminando ($cn - cn\log n$) maggioro:

$$cn\,log^2 n - 2cn\log n + cn + an\log n \le cn\,log^2 n - cn\log n + an\log n$$

Inoltre

$$cn\,log^2 n - cn\log n + an\log n \le cn\,log^2 n \text{ sse } c \ge a.$$

Concludiamo che

$$\boldsymbol{T(n) = O(n\,log^2 n)}$$

Si lascia per esercizio dimostrare che $T(n) = \Omega(n\log^2 n)$

Esercizio 7 – dallo pseudocodice all'equazione

```
funzione Palindromo_Ric (A: vettore; in,fi: intero)
1)    if fi – in ≤ 1 return TRUE
2)    if A[in] = A[fi] return FALSE
3)    return Palindromo_Ric(A; in+1, fi−1)
```

1) $\Theta(1)$ – caso base

1,2) $\Theta(1)$ – parte al di fuori della chiamata ricorsiva

3) $T(n-2)$ – una chiamata ricorsiva

Equazione:

$$T(n) = T(n-2) + \Theta(1)$$

$$T(1) = \Theta(1)$$

Esercizio 8 – dallo pseudocodice all'equazione

```
Funzione Test (n: intero)
1)    k ← 0
2)    for i = 1 to n do
2)          k ← k + 1
3)    if n ≤ 1 return k
4)    else return (Test(n DIV 2)+Test(n DIV 4))
```

1) $\Theta(1)$ – parte al di fuori della chiamata ricorsiva
2) $\Theta(n)$ – iterazione, parte al di fuori della chiamata ricorsiva
3) $\Theta(1)$ – caso base
4) $T(n/2) + T(n/4)$ – due chiamate ricorsive

Equazione:

$$T(n) = T(n/2) + T(n/4) + \Theta(n)$$
$$T(1) = \Theta(1)$$

Nota: DIV fa la divisione intera

Esercizio 9 – dallo pseudocodice all'equazione

```
Funzione Fun (n: intero)
1)     k ← 1
2)     while k ≤ n do
2)            k ← 2*k
3)     if n ≤ 1 return k
4)     else return (Fun(n DIV 2)))
```

ATTENZIONE!

1) $\Theta(1)$ – parte al di fuori della chiamata ricorsiva
2) $\Theta(\log n)$ – iterazione, parte al di fuori della chiamata ricorsiva
3) $\Theta(1)$ – caso base
4) $T(n/2)$ – una chiamata ricorsiva

Equazione:

$$T(n) = T(n/2) + \Theta(\log n)$$

$$T(1) = \Theta(1)$$

Nota: DIV fa la divisione intera