Corso di laurea in Informatica
Introduzione agli Algoritmi

Esercizi sulle Equazioni di ricorrenza

Angelo Monti

SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio 1

Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n \leq 1 \\ T\left(\frac{n}{2}\right) + \Theta(1) & \text{altrimenti} \end{cases}$$

Metodo iterativo

Espandiamo la ricorrenza iterativamente:

$$
\begin{aligned}
T(n) &= T\left(\frac{n}{2}\right) + \Theta(1) \\
&= \left(T\left(\frac{n}{4}\right) + \Theta(1)\right) + \Theta(1) \\
&= T\left(\frac{n}{4}\right) + 2\Theta(1) \\
&= \left(T\left(\frac{n}{8}\right) + \Theta(1)\right) + 2\Theta(1) \\
&= T\left(\frac{n}{8}\right) + 3\Theta(1).
\end{aligned}
$$

Proseguendo per k passi, otteniamo:

$$T(n) = T\left(\frac{n}{2^k}\right) + k\Theta(1).$$

$$T(n) = T\left(\frac{n}{2^k}\right) + k\Theta(1).$$

Quando k è tale che $\frac{n}{2^k} = 1$, cioè $k = \log_2 n$, raggiungiamo la condizione iniziale $T(1) = \Theta(1)$, quindi:

$$T(n) = T(1) + \log_2 n \cdot \Theta(1).$$

Poiché $T(1) = \Theta(1)$, possiamo scrivere:

$$T(n) = \Theta(1) + \log_2 n \cdot \Theta(1).$$

Semplificando, otteniamo:

$$T(n) = \log_2 n \cdot \Theta(1).$$

Poiché $\log_2 n = \Theta(\log n)$, la soluzione asintotica finale è:

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

Metodo dell'albero

$$T(n) = \begin{cases} \Theta(1) & \text{se } n \leq 1 \\ T\left(\frac{n}{2}\right) + \Theta(1) & \text{altrimenti} \end{cases}$$

In questa equazione di ricorrenza è presente un'unica chiamata ricorsiva. Di conseguenza, l'albero di ricorsione si riduce a una semplice catena, e il metodo di soluzione basato sull'albero si riconduce al metodo iterativo.

Metodo di sostituzione

$$T(n) = \begin{cases} \Theta(1) & \text{se } n \leq 1 \\ T\left(\frac{n}{2}\right) + \Theta(1) & \text{altrimenti} \end{cases}$$

Supponiamo che $T(n) = \Theta(\log n)$.

Dimostreremo che $T(n) = \Theta(\log n)$ in due passi: prima mostrando che $T(n) \in O(\log n)$, e successivamente dimostrando che $T(n) \in \Theta(\log n)$.

Passo 1: Dimostrare che $T(n) \in O(\log n)$

Eliminando l'asintotica dall'equazione di ricorrenza e partendo per semplicità di calcoli da $T(2)$, otteniamo:

$$T(n) \leq \begin{cases} b & \text{se } n \leq 2 \\ T\left(\frac{n}{2}\right) + a & \text{altrimenti} \end{cases}$$

Dimostreremo che esiste una costante c per cui $T(n) \leq c \log_2 n$. Prendendo $c \geq b$ abbiamo

$$T(2) \leq b \leq c \cdot 1 = c \log_2 2$$

e l'ipotesi vale per il caso base.
Applicando l'ipotesi di induzione, cioè $T\left(\frac{n}{2}\right) \leq c \log_2 \frac{n}{2}$), otteniamo:

$$T(n) \leq c(\log_2 n - 1) + a = c \log_2 n - (c - a) \leq c \log_2 n$$

Dove l'ultima diseguaglianza vale prendendo $c \geq a$.
Quindi, prendendo $c = \max\{a, b\}$ abbiamo dimostrato che:

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

Passo 2: Dimostrare che $T(n) \in \Omega(\log n)$

Eliminando l'asintotica dall'equazione di ricorrenza, otteniamo:

$$T(n) \geq \begin{cases} b & \text{se } n \leq 1 \\ T\left(\frac{n}{2}\right) + a & \text{altrimenti} \end{cases}$$

Dimostreremo che esiste una costante c per cui $T(n) \geq c \log_2 n$. Abbiamo

$$T(1) \geq b \geq 0 = c \cdot \log_2 1$$

e l'ipotesi vale per il caso base.
Supponiamo che l'ipotesi di induzione sia valida, cioè che

$$T\left(\frac{n}{2}\right) \geq c \log_2 \frac{n}{2}.$$

Sostituendo questa ipotesi nella ricorrenza, otteniamo:

$$T(n) \geq c(\log_2 n - 1) + a = c \log_2 n + (a - c) \geq c \log_2 n$$

Dove l'ultima diseguaglianza vale prendendo $c \leq a$.
Quindi, prendendo $c = a$ abbiamo dimostrato che:

$$T(n) = \Omega(\log_2 n)$$

Teorema principale

$$T(n) = \begin{cases} \Theta(1) & \text{se } n \leq 1 \\ T\left(\frac{n}{2}\right) + \Theta(1) & \text{altrimenti} \end{cases}$$

Calcoliamo $\log_b a = \log_2 1 = 0$.
Poiché $f(n) = \Theta(1) = \Theta(n^{\log_b a})$, il caso 2 si applica.
La soluzione è:

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

Esercizio 2

Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 2 \\ 2T(\sqrt{n}) + \Theta(\log n) & \text{altrimenti} \end{cases}$$

Metodo iterativo

Espandiamo la ricorrenza iterativamente:

$$
\begin{aligned}
T(n) &= 2T\left(n^{\frac{1}{2}}\right) + \Theta(\log n) \\
&= 2\left(2T\left(\left(n^{\frac{1}{2}}\right)^{\frac{1}{2}}\right) + \Theta\left(\log n^{\frac{1}{2}}\right)\right) + \Theta(\log\ n) \\
&= 2^2T\left(n^{\frac{1}{2^2}}\right) + 2\cdot\Theta(\frac{1}{2}\log n) + \Theta(\log\ n) \\
&= 2^2T\left(n^{\frac{1}{2^2}}\right) + 2\cdot\Theta(\log n) \\
&= \ldots\ldots \\
&= 2^iT\left(n^{\frac{1}{2^i}}\right) + i\cdot\Theta(\log n)
\end{aligned}
$$

$$2^i T\left(n^{\frac{1}{2^i}}\right) + i \cdot \Theta(\log n)$$

l'iterazione termina quando $n^{\frac{1}{2^i}} = 2$, ovvero quando $\frac{1}{2^i}\log_2 n = 1$. Da questo otteniamo $\log_2 n = 2^i$, il che implica che $i = \log_2 \log_2 n$.
A quel punto, otteniamo:

$$\begin{aligned}
T(n) &= 2^{\log_2 \log_2 n} T(2) + \log_2 \log_2 n \Theta(\log n) \\
&= \Theta(\log_2 n) + \Theta(\log n \log \log n) \\
&= \Theta(\log n \log \log n)
\end{aligned}$$

Metodo dell'albero

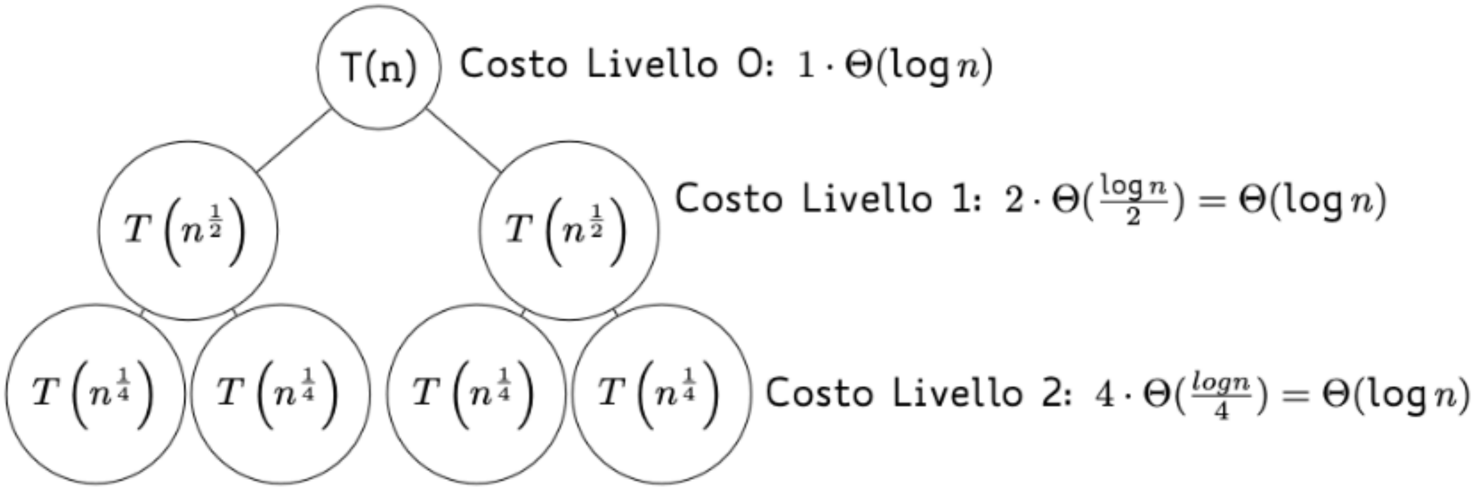
$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 2 \\ 2T(\sqrt{n}) + \Theta(\log n) & \text{altrimenti} \end{cases}$$

Costruzione dell'albero.

- Al livello 0, il costo è $f(n) = \Theta(\log n)$.
- Al livello 1, abbiamo 2 sottoproblemi di dimensione $\sqrt{n}$, ciascuno con costo $\Theta\left(\log n^{\frac{1}{2}}\right) = \Theta(\frac{\log n}{2})$, per un costo totale di $2 \cdot \Theta(\frac{\log n}{2}) = \Theta(\log n)$.
- Al livello k, ci sono 2^k sottoproblemi di dimensione $n^{\frac{1}{2^k}}$, ciascuno con costo $\Theta\left(\frac{\log n}{2^k}\right)$, per un costo totale di:

$$\text{Costo livello } k = 2^k \cdot \Theta\left(\frac{\log n}{2^k}\right) = \Theta(\log n).$$

Nell'immagine che segue ci sono i primi livelli dell' albero di ricorsione per $T(n) = 2T(\sqrt{n}) + \Theta(\log n)$:



Calcolo del numero di livelli. L'albero si espande fino a quando i sottoproblemi raggiungono dimensione 2 cioé:

$$n^{\frac{1}{2^k}} = 2 \Rightarrow k = \log_2 \log_2 n$$

Costo complessivo. Il costo totale è la somma dei costi su tutti i livelli:

$$\text{Costo totale} = \sum_{k=0}^{\log_2 \log_2 n} \Theta(\log n) = \Theta(\log n \log \log n).$$

Metodo di sostituzione

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 2 \\ 2T(\sqrt{n}) + \Theta(\log n) & \text{altrimenti} \end{cases}$$

Dimostreremo che $T(n) = \Theta(\log n \log \log n)$ in due passi:

1. prima mostrando che $T(n) \in O(\log n \log \log n)$,
2. e successivamente dimostrando che $T(n) \in \Omega(\log n \log \log n)$.

Passo 1: Dimostrare che $T(n) \in O(\log n \log\log n)$

Eliminando l'asintotica dall'equazione di ricorrenza e partendo per semplicità di calcoli da $n = 4$, otteniamo:

$$T(n) = \begin{cases} b & \text{se } n = 4 \\ 2T(\sqrt{n}) + a\log_2 n & \text{altrimenti} \end{cases}$$

Dimostreremo che esiste una costante c per cui

$$T(n) \leq c \log_2 n \log_2 \log_2 n.$$

Prendendo $c \geq b$ e considerando che $\log_2 4 \log_2 \log_2 4 = 2 > 0$ abbiamo

$$T(4) = b \leq c \log_2 4 \log_2 \log_2 4$$

e l'ipotesi vale per il caso base.

Sostituendo l'ipotesi induttiva per $T(\sqrt{n})$, otteniamo:

$$T(n) \leq 2c \cdot \log_2 \sqrt{n} \cdot \log_2 \log_2 \sqrt{n} + a\log_2 n.$$

Poiché $\log_2 \sqrt{n} = \frac{\log_2 n}{2}$ e $\log_2 \log_2 \sqrt{n} = \log_2 \log_2 n - 1$, abbiamo:

$$T(n) \leq c \cdot \log_2 n \log_2 \log_2 n - c\log_2 n + a\log_2 n \leq c \cdot \log_2 n \log_2 \log_2 n.$$

dove l'ultima diseguaglianza vale se $c \geq a$. Quindi, prendendo $c = \max\{a, b\}$, abbiamo dimostrato che:

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

Passo 2: Dimostrare che $T(n) \in \Omega(\log n \log\log n)$

Eliminando l'asintotica dall'equazione di ricorrenza otteniamo:

$$T(n = \begin{cases} b & \text{se } n = 2 \\ 2T(\sqrt{n}) + a\log_2 n & \text{altrimenti} \end{cases}$$

Per dimostrare che $T(n) \in \Omega(\log n \log\log n)$, possiamo utilizzare un argomento analogo a quello che abbiamo usato per la parte di $O(\log n \log\log n)$, ma in senso opposto. Ipotizziamo che esista una costante $c > 0$ tale che:

$$T(n) \geq c \log_2 n \log_2 \log_2 n$$

Per $n = 2$ abbiamo

$$T(2) = b \geq 0 = c \cdot 0 = c \log_2 2 \log_2 \log_2 2$$

dove l'ultimo passaggio segue perché $\log_2 \log_2 2 = 0$. Dunque l'ipotesi vale per il caso base.

Supponendo che per $T(\sqrt{n})$ valga l'ipotesi induttiva:

$$T(\sqrt{n}) \geq c \log_2 \sqrt{n} \log_2 \log_2 \sqrt{n}.$$

Poiché $\log_2 \sqrt{n} = \frac{\log_2 n}{2}$ e $\log_2 \log_2 \sqrt{n} = \log_2 \log_2 n - 1$, abbiamo:

$$T(\sqrt{n}) \geq c \cdot \frac{\log_2 n}{2} \cdot (\log_2 \log_2 n - 1).$$

Sostituendo nella ricorrenza:

$$T(n) \geq 2 \cdot \left(c \cdot \frac{\log_2 n}{2} \cdot (\log_2 \log_2 n - 1)\right) + a \log_2 n.$$

Semplificando:

$$T(n) \geq c \cdot \log_2 n \log_2 \log_2 n + (a - c) \log_2 n \geq d \cdot \log_2 n \cdot \log_2 \log_2 n$$

dove l'ultima diseguaglianza segue se $c \leq a$.
Quindi, assumendo $c = a$, possiamo concludere che : $T(n) \in \Omega(\log n \log\log n)$.

Esercizio 3
Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 4T\left(\frac{n}{2}\right) + \Theta(n) & \text{altrimenti} \end{cases}$$

Metodo iterativo

$$\begin{aligned}
T(n) &= 4T\left(\frac{n}{2}\right) + \Theta(n) \\
&= 4\left(4T\left(\frac{n}{2^2}\right) + \Theta\left(\frac{n}{2}\right)\right) + \Theta(n) \\
&= 4^2T\left(\frac{n}{2^2}\right) + 4\Theta\left(\frac{n}{2}\right) + \Theta(n) \\
&= \ldots\ldots \\
&= 4^iT\left(\frac{n}{2^i}\right) + \sum_{j=0}^{i-1} 4^j\Theta\left(\frac{n}{2^j}\right)
\end{aligned}$$

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

Ci fermiamo quando $\dfrac{n}{2^i} = 1$ vale a dire $i = \log_2 n$ ed otteniamo ....

$$\begin{aligned}
T(n) &= 4^{\log_2 n} T(1) + \Theta\left(n \sum_{j=0}^{\log n - 1} 2^j\right) \\
&= 2^{2\log_2 n}\Theta(1) + \Theta\left(n 2^{\log n}\right) \\
&= \Theta(n^2) + \Theta\left(n^2\right) \\
&= \Theta\left(n^2\right)
\end{aligned}$$

Dunque la soluzione tramite questo metodo è $\Theta(n^2)$.

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 4T\left(\frac{n}{2}\right) + \Theta(n) & \text{altrimenti} \end{cases}$$

Metodo principale

Possiamo applicare questo metodo perché la ricorrenza soddisfa le ipotesi del teorema; inoltre:

- $a = 4, \;\; b = 2$
- $n^{\log_b a} = n^{\log_2 4} = n^2$
- $f(n) = \Theta(n) = O\left(n^{\log_b a - \epsilon}\right)$ (ad es. per $\epsilon = 1$)

Siamo quindi nel **caso 1**, da cui: $T(n) = \Theta(n^2)$.

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 4T\left(\frac{n}{2}\right) + \Theta(n) & \text{altrimenti} \end{cases}$$

Metodo di sostituzione

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

$$T(n) = 4T\left(\frac{n}{2}\right) + b \cdot n$$

$$T(1) = a$$

Dove a e b sono costanti positive. Proviamo a dimostrare per induzione che la soluzione è:

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

Passo base. $T(1) = a \geq c$, da cui deduciamo una prima condizione su c.

Passo induttivo. Sostituendo nell'equazione generica otteniamo:

$$
\begin{aligned}
T(n) \quad & \geq \quad 4c\left(\frac{n}{2}\right)^2 + b \cdot n \\
& = \quad c \cdot n^2 + b \cdot n \\
& \geq \quad c \cdot n^2
\end{aligned}
$$

Ne concludiamo che $T(n) = \Omega(n^2)$.

Per quanto riguarda la maggiorazione, tentiamo la soluzione $T(n) \leq c \cdot n^2$ dove c è una costante da determinare.

Passo base. $T(1) = a \leq c$, da cui deduciamo una prima condizione su c.
Passo induttivo. Sostituendo nell'equazione generica otteniamo:

$T(n) \leq 4c\left(\frac{n}{2}\right)^2 + b \cdot n = c \cdot n^2 + b \cdot n$ che non è mai $\leq c \cdot n^2$ perché b è una costante positiva.

Tentiamo allora $T(n) \leq c \cdot n^2 - d \cdot n$.

Passo base. $T(1) = a \leq c - d$ che è vera per certi valori di c e d (*ad esempio* $c = 2d$ *e* $d \geq a$)

Passo induttivo. Sostituendo l'ipotesi induttiva nell'equazione otteniamo:

$$\begin{aligned} T(n) &\leq 4\left(c\left(\frac{n}{2}\right)^2 - d\left(\frac{n}{2}\right)\right) + b \cdot n \\ &= c \cdot n^2 - 2d \cdot n + b \cdot n \\ &= c \cdot n^2 - d \cdot n - d \cdot n + b \cdot n \\ &\leq c \cdot n^2 - d \cdot n \end{aligned}$$

Dove l'ultima diseguaglianza segue prendendo $d \geq b$.

Ne concludiamo che prendendo $c = 2d$e $d = \max\{a, b\}$si ha

$$T(n) = O(n^2)\,.$$

*U*nendo le due limitazioni abbiamo: $T(n) = \Theta(n^2)\,.$

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 4T\left(\frac{n}{2}\right) + \Theta(n) & \text{altrimenti} \end{cases}$$

Metodo dell'albero

- *Nell'albero ogni nodo ha 4 figli questo significa che a livello i ci saranno 4^i nodi ciascuno dei quali contribuirà per un costo* $\Theta\left(\frac{n}{2^i}\right)$.
- *La ricorsione termina quando* $\frac{n}{2^i} = 1$, *a dire* $i = \log_2 n$ *questo significa che i livelli dell'albero sono* $\log n + 1$ *(da 0 a* $\log n$*).*

Il contributo totale sommato per livelli sarà

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

Dove per risolvere la sommatoria ho utilizzato $\sum_{i=0}^{x} 2^i = \Theta\left(2^x\right)$

Esercizio 4

Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{altrimenti} \end{cases}$$

Metodo Iterativo

$$
\begin{aligned}
T(n) &= 2T\left(\frac{n}{2}\right) + \Theta(n^2) \\
&= 2\left(2T\left(\frac{n}{2^2}\right) + \Theta\left(\left(\frac{n}{2}\right)^2\right)\right) + \Theta(n^2) \\
&= 2^2T\left(\frac{n}{2^2}\right) + 2\Theta\left(\left(\frac{n}{2}\right)^2\right) + \Theta(n^2) \\
&= \ldots\ldots \\
&= 2^iT\left(\frac{n}{2^i}\right) + \sum_{j=0}^{i-1} 2^j\Theta\left(\left(\frac{n}{2^j}\right)^2\right) \\
&= 2^iT\left(\frac{n}{2^i}\right) + \Theta\left(n^2\sum_{j=0}^{i-1}\frac{1}{2^j}\right)
\end{aligned}
$$

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

Ci fermiamo quando $\frac{n}{2^i} = 1$ vale a dire $i = \log_2 n$ ed otteniamo ....

$$\begin{aligned}
T(n) &= 2^{\log_2 n} T(1) + \Theta\left(n^2 \sum_{j=0}^{\log n - 1} \frac{1}{2^j}\right) \\
&= n\Theta(1) + \Theta(n^2)\Theta(1) \\
&= \Theta(n) + \Theta(n^2) \\
&= \Theta\left(n^2\right)
\end{aligned}$$

Dove ho utilizzato $\sum_{i=0}^{x} c^i = \Theta(1)$ quando $c < 1$

Dunque la soluzione tramite questo metodo è $\Theta(n^2)$.

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{altrimenti} \end{cases}$$

Metodo Principale

Possiamo applicare questo metodo perché la ricorrenza soddisfa le ipotesi del teorema; inoltre:

- $a = 2,\ b = 2$
- $n^{\log_b a} = n^{\log_2 2} = n$
- $f(n) = \Theta(n^2) = \Omega(n^{\log_b a+\epsilon})$ (ad es. per ε=1)

Poiché $a \cdot f\left(\dfrac{n}{b}\right) = 2\left(\dfrac{n}{2}\right)^2 = \dfrac{n^2}{2} < cn^2$ con $c = \dfrac{1}{2}$, siamo nel caso 3 e possiamo concludere che $T(n) = \Theta(n^2)$

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{altrimenti} \end{cases}$$

Metodo di sostituzione

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

$$T(n) = 2T\left(\frac{n}{2}\right) + b \cdot n^2$$

$$T(1) = a$$

Per a e b costanti positive.

Proviamo a dimostrare per induzione la soluzione:

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

Passo base. $T(1) \le a \le c$, da cui deduciamo una prima condizione su c.

Passo induttivo. Sostituendo nell'equazione generica otteniamo:

$$\begin{aligned} T(n) &\le 2c\left(\frac{n}{2}\right)^2 + b \cdot n^2 \\ &= c\frac{n^2}{2} + b \cdot n^2 \\ &= \left(\frac{c}{2} + b\right) n^2 \\ &\le c \cdot n^2 \end{aligned}$$

Dove l'ultima disuguaglianza vale se $\frac{c}{2} + b \le c$ e basta quindi prendere $c \ge 2b$.

Abbiamo dimostrato che $T(n) = O(n^2)$

Proviamo ora a dimostrare per induzione che vale anche $T(n) = \Omega(n^2)$. Ipotizziamo dunque:

$T(n) \geq c \cdot n^2$ dove c è una costante da determinare.

Passo base. $T(1) = a \geq c$, da cui deduciamo una prima condizione su c.

Passo induttivo. Sostituendo nell'equazione generica otteniamo:

$$
\begin{aligned}
T(n) &\geq 2c\left(\frac{n}{2}\right)^2 + b \cdot n^2 \\
&= c\frac{n^2}{2} + b \cdot n^2 \\
&= \left(\frac{c}{2} + b\right) n^2 \\
&\geq c \cdot n^2
\end{aligned}
$$

Dove l'ultima disuguaglianza vale se $\frac{c}{2} + b \geq c$ e basta quindi prendere $c \leq 2b$.

Ne deduciamo che con $c = \min\{a, 2b\}$ vale $T(n) = \Omega(n^2)$

Dalle due limitazioni concludiamo che $T(n) = \Theta(n^2)$

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n^2) & \text{altrimenti} \end{cases}$$

Metodo dell'albero

- *Nell'albero ogni nodo ha 2 figli questo significa che a livello* i *ci saranno* 2^i *nodi ciascuno dei quali contribuirà per un costo* $\Theta\left(\left(\frac{n}{2^i}\right)^2\right)$.
- *La ricorsione termina quando* $\frac{n}{2^i} = 1$, *vale a dire* $i = \log_2 n$ *questo significa che i livelli dell'albero sono* $\log n + 1$ *(da 0 a* $\log n$*).*

Il contributo totale sommato per livelli sarà

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

Dove per la sommatoria ho utilizzato $\sum_{i=0}^{x} c^i = \Theta(1)$ *quando* $c < 1$.

Esercizio 5
Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n \log n) & \text{altrimenti} \end{cases}$$

Metodo principale:

- $a = b = 2$
- $n^{\log_b a} = n$
- n è asintoticamente più piccolo di $f(n) = n \log n$, ma non polinomialmente più piccolo.

Di conseguenza **non** possiamo applicare il metodo del teorema principale.

Metodo iterativo

$$
\begin{aligned}
T(n) &= 2T\left(\frac{n}{2}\right) + \Theta(n \log n) \\
&= 2\left(2T\left(\frac{n}{2}\right) + \Theta\left(\frac{n}{2}\log\frac{n}{2}\right)\right) + \Theta(n \log n) \\
&= 2^2 T\left(\frac{n}{2^2}\right) + \Theta\left(n \log \frac{n}{2}\right) + \Theta(n \log n) \\
&= \ldots\ldots \\
&= 2^i T\left(\frac{n}{2^i}\right) + \Theta\left(\sum_{j=0}^{i-1} n \log \frac{n}{2^j}\right) \\
&= 2^i T\left(\frac{n}{2^i}\right) + \Theta\left(n \sum_{j=0}^{i-1} (\log n - \log 2^j)\right) \\
&= 2^i T\left(\frac{n}{2^i}\right) + \Theta\left(n \sum_{j=0}^{i-1} (\log n - j)\right)
\end{aligned}
$$

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

Ci fermiamo quando $\frac{n}{2^i} = 1$ vale a dire $i = \log_2 n$ ed otteniamo ....

$$\begin{aligned}
T(n) &= 2^{\log_2 n} T(1) + \Theta\left(n \sum_{j=0}^{\log n - 1} (\log n - j)\right) \\
&= n\Theta(1) + \Theta\left(n \sum_{k=1}^{\log n} k\right) \\
&= \Theta(n) + \Theta\left(n \log^2 n\right) \\
&= \Theta\left(n \log^2 n\right)
\end{aligned}$$

Dove ho usato che $\sum_{k=1}^{x} k = \Theta\left(x^2\right)$

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n \log n) & \text{altrimenti} \end{cases}$$

Metodo di sostituzione

Impostiamo la dimensione del caso base a 2, per evitare di dover gestire il caso di log 1 = 0 ➟ T(2)= $\Theta(1)$.

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

- $T(n) = 2T\left(\dfrac{n}{2}\right) + b \cdot n \log_2 n$
- $T(2) = a$

dove a e b sono costanti positive

Proviamo a dimostrare per induzione la soluzione:

$T(n) \leq cn\log_2^2 n$, dove c è una costante da determinare.

Passo base. $T(2) = a \leq c * 2 * 1$, che è vera per $c \geq a/2$.

Passo induttivo:

$$
\begin{aligned}
T(n) &= 2T\left(\frac{n}{2}\right) + b \cdot n\log_2 n \\
&\leq 2c \cdot \frac{n}{2}\log_2^2 \frac{n}{2} + b \cdot n\log_2 n \\
&\leq c \cdot n(\log_2 n - 1)^2 + b \cdot n\log_2 n \\
&\leq c \cdot n(\log_2 n + 1 - 2\log_2 n) + b \cdot n\log_2 n \\
&\leq c \cdot n\log_2 n + c \cdot n - 2c \cdot n\log_2 n + b \cdot n\log_2 n \\
&\leq c \cdot n\log_2 n - c \cdot n\log_2 n + b \cdot n\log_2 n \\
&\leq c \cdot n\log_2 n
\end{aligned}
$$

Dove la penultima diseguaglianza vale perché $cn \leq cn\log n$ e per l'ultima diseguaglianza basta prendere $c \geq b$

Abbiamo quindi dimostrato che $T(n) = O(n\log^2 n)$

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

Esercizio 6

Consideriamo la seguente ricorrenza:

$$T(n) = \begin{cases} \Theta(1) & \text{se } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta\left(\frac{n}{\log n}\right) & \text{altrimenti} \end{cases}$$

Metodo principale:

- $a = b = 2$
- $n^{\log_b a} = n$
- n è asintoticamente più grande di $f(n) = \dfrac{n}{\log n}$, ma non polinomialmente.

Di conseguenza **non** possiamo applicare il metodo del teorema principale.

Metodo iterativo

$$
\begin{aligned}
T(n) &= 2T\left(\frac{n}{2}\right) + \Theta\left(\frac{n}{\log n}\right) \\
&= 2\left(2T\left(\frac{n}{2}\right) + \Theta\left(\frac{n/2}{\log n/2}\right)\right) + \Theta\left(\frac{n}{\log n}\right) \\
&= 2^2T\left(\frac{n}{2^2}\right) + \Theta\left(\frac{n}{\log_2 n/2}\right) + \Theta\left(\frac{n}{\log n}\right) \\
&= \ldots\ldots \\
&= 2^iT\left(\frac{n}{2^i}\right) + \Theta\left(\sum_{j=0}^{i-1} \frac{n}{\log \frac{n}{2^j}}\right) \\
&= 2^iT\left(\frac{n}{2^i}\right) + \Theta\left(n\sum_{j=0}^{i-1} \frac{1}{\log n - j}\right)
\end{aligned}
$$

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

Ci fermiamo quando $\dfrac{n}{2^i} = 1$ vale a dire $i = \log_2 n$ ed otteniamo ....

$$\begin{aligned}
T(n) &= 2^{\log_2 n} T(1) + \Theta\left(n \sum_{j=0}^{\log n - 1} \frac{1}{\log n - j}\right) \\
&= n\Theta(1) + \Theta\left(n \sum_{k=1}^{\log n} \frac{1}{k}\right) \\
&= \Theta(n) + \Theta\left(n \log\log n\right) \\
&= \Theta\left(n \log\log n\right)
\end{aligned}$$

Dove ho usato che $\displaystyle\sum_{k=1}^{x} \frac{1}{k} = \Theta(\log x)$