Homework 1

Alessandro Falino, Nike di Giacomo

October 24, 2023

1 Esercizio 1.1

1. f2 - f1 - f4 - f3
2. f1 - f4 - f3 - f2
3. f4 - f1 - f2 - f3

2 Esercizio 1.2

1. g(n) = O(f(n))
2. f(n) = O(g(n))
3. f(n) = O(g(n))
4. g(n) = O(f(n))
5. g(n) = O(f(n))
6. f(n) = O(g(n))

3 Esercizio 1.3

1. vero
2. falso

4 Esercizio 1.4

1. 2T(n/2) + $O(\sqrt{n})$

   Per un nodo al livello i, la dimensione dei sottoproblem è $n/2^i$. La dimensione 1 si raggiunge per

   $$n/2^i = 1 => i = log_2(n)$$

   L'altezza di questo albero è quindi $log_2(n) + 1$. Ciascun livello ha il doppio dei nodi di quello precedente, per cui al livello i-esimo avremo 2^i nodi. Dato quindi il costo di un singolo nodo pari ad $c(\sqrt{n}/2)$:

   $$T(n) = \sum_{i=0}^{log_2(n)-1} 2^i * c(\frac{n}{2^i})^{0.5} + 2^{log_2(n)T(1)}$$

   $$\sum_{i=0}^{log_2(n)-1} 2^i * c(\frac{n}{2^i})^{0.5} + \Theta(n^{log_2(2)})$$

Sapendo che:

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

Possiamo riscrivere il risultato precedente come:

$$\sum_{i=0}^{log_2(n)-1} c(\frac{2}{\sqrt{2^i}})^i * \sqrt{n} + \Theta(n^{log_2(2)})$$

$$=$$

$$\frac{(\frac{2}{\sqrt{2}})^{log_2(n)}-1}{0.4} + \Theta(n)$$

Possiamo approssimare il $(\frac{2}{\sqrt{2}})$ ad 1, per cui per ogni numero n molto alto otteniamo 1-1, che non influisce sul risultato. Possiamo quindi dire che il risultato è $\Theta(n)$

Verifichiamo questo risultato con il teorema dell'esperto: a=2, b=2, $f(n)$=$O(\sqrt{n})$.
Per questi valori otteniamo quindi che:

$$f(n) = O(n^{log_b a-\epsilon})$$

Per $\epsilon = 2-\sqrt{2}$, otteniamo proprio:

$$f(n) = O(\sqrt{n}) = O(n^{0.5})$$

Questo rientra nel caso 1 del teorema dell'esperto per cui:

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

2. T(n)=T($\sqrt{n}$) + $\Theta(log(log(n)))$

   Ponendo $m = log_2(n)$, avremo $n = 2^m$, per cui:

   $$S(n) = S(m/2) + \Theta(log(m))$$

   L'altezza di questo albero è $log_2(m)$, il costo per livello è $\Theta(log_2(m))$. Il costo totale è quindi:

   $$log_2(m) * \Theta(log_2(m))$$

   Questo equivale quindi a:

   $$\Theta(log_2(m) * log_2(m))$$

   Sostituiamo m con n, ed otteniamo:

   $$\Theta(log_2 log_2(n) * log_2 log_2(n)) = \Theta(log_2^2(log_2(n)))$$

   Questa ricorrenza non può essere risolta con il teorema dell'esperto.

3. T(n)=10T(n/3)+ $17n^{1.2}$

   Per un nodo al livello i, la dimensione dei sottoproblemi è $n/3^i$. La dimensione 1 si raggiunge per

   $$n/3^i = 1 => i = log_3(n)$$

   L'altezza di questo albero è quindi $log_3(n) + 1$. Ciascun livello ha il 10 volte i nodi di quello precedente, per cui al livello i-esimo avremo 10^i nodi. Dato quindi il costo di un singolo nodo pari ad $c(n/3^i)^{1.2}$:

   $$T(n) = \sum_{i=0}^{log_3(n)-1} 10^i * c(\frac{n}{3^i})^{1.2} + 10^{log_3(n)T(1)}$$

$$\sum_{i=0}^{log_3(n)-1} 10^i * c(\frac{n}{3^i})^{1.2} + \Theta(n^{log_3(10)})$$

Sapendo che:

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

Possiamo riscrivere il risultato precedente come:

$$\sum_{i=0}^{log_2(n)-1} (\frac{10}{3^{1.2}})^i * c * n^{1.2} + \Theta(n^2)$$

$$=$$

$$(\frac{3.73^{log_3(n)}-1}{2.67-1}) * c * n^{1.2} + \Theta(n^2)$$

Possiamo trascurare i valori di ordini di grandezza inferiori e dire quindi che il risultato è $\Theta(n^2)$

Verifichiamo questo risultato con il teorema dell'esperto: a=10, b=3, $f(n)$=17$n^{1.2}$.
Per questi valori otteniamo quindi che:

$$n^{1.2} = O(n^{log_b a - \epsilon})$$

Per $\epsilon \approx 6.2681$, otteniamo proprio:

$$n^{1.2} = n^{1.2}$$

Questo rientra nel caso 1 del teorema dell'esperto per cui:

$$T(n) = \Theta(n^{log_b a}) = \Theta(n^{log_3 10}) \approx \Theta(n^2)$$

4. T(n) = T(n/3) + T(2n/3) + cn

   L'altezza di questo albero è data da $\frac{n}{3^i} = 1$ da cui $i = log_3(n)$ ed $n(\frac{2}{3})^i = 1$ da cui $i = log_{3/2}(n)$
   Il costo di un singolo nodo è distinguibile in: $c(\frac{n}{3^i})$ e $cn(\frac{2}{3})^i$ pertanto

$$nlog_3(n) \leq T(n) \leq nlog_{3/2}(n)$$

   Quindi essendo T(n) caratterizzata da l limite inferiore $nlog_3(n)$, possiamo dire che $\Omega(nlog_3(n))$