Esame del 08.07.2024

Algoritmi e Laboratorio

Parte B

Esercizio 1. Si consideri un albero rosso-nero con n nodi interni in cui $bh(x)$ denota l'altezza nera di un nodo x.

A. Si dimostri che il sottoalbero radicato a un nodo x contiene almeno $2^{bh(x)} - 1$ nodi interni. (Suggerimento: fare una dimostrazione per induzione sull'altezza di x.)

B. Utilizzando il risultato del punto **A**, si dimostri che l'altezza dell'albero rosso-nero è al più $2\log(n+1)$.

C. Come possiamo scrivere il valore dell'altezza in notazione asintotica, in base al risultato del punto **B**?

Esercizio 2. **A**. Si utilizzi il metodo Master per risolvere la sequente equazione di ricorrenza

$$T(n) = T\left(\frac{2}{3}n\right) + \Theta(1). \tag{1}$$

B. L'equazione (1) rappresenta un limite superiore al tempo computazionale di Heapify, ovvero $T_{\text{Heapify}}(n) \leq T_{\text{Heapify}}\left(\frac{2}{3}n\right) + \Theta(1)$. Come possiamo allora scrivere il tempo computazionale di Heapify in notazione asintotica?

C. Si disegni l'albero di ricorsione di

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

indicando **il costo dell i-esimo livello, l'altezza e il numero di foglie**. Si usino i risultati per stimare (ovvero fare una guess per) l'ordine di grandezza della soluzione all'equazione (2) in notazione asintotica.

Soluzioni

Esercizio 1. **A**. Se $h(x) = 0$, allora x è una foglia, dunque il sottoalbero radicato a x contiente 0 nodi, $bh(x) = 0$ e pertanto $2^{bh(x)} - 1 = 2^0 - 1 = 0$, ovvero la relazione è vera per il caso base. Supponiamo la tesi vera per tutti i nodi di altezza $0 \leq h < h(x)$.

Poichè $h(x) > 0$, il nodo x ha due figli, che indicheremo con x.child, ognuno dei queli può essere nero, e avere altezza nera $bh(\text{x.child}) = bh(x) - 1$, oppure rosso, e avere altezza nera $bh(\text{x.child}) = bh(x)$;in ogni caso, $bh(\text{x.child}) \geq bh(x) - 1$. Inoltre, $h(\text{x.child}) < h(x)$ e i sottoalberi radicati ai nodi figli di x ha almeno $2^{bh(\text{x.child})} - 1 \geq 2^{bh(x)-1} - 1$ nodi interni. Per concludere la prova, basta osservare che il numero di nodi interni al sottoalbero radicato ad x è $\geq 2(2^{bh(x)-1} - 1) + 1 = 2^{bh(x)} - 1$.

B. Sia $h^\star$ l'altezza dell'albero. Poichè almeno la metà dei nodi su un cammino dalla radice alle foglie è fatta di nodi neri, si ha che $bh(\text{root}) \geq \frac{h^\star}{2}$ e quindi $n \geq 2^{bh(\text{root})} - 1 \geq 2^{\frac{h^\star}{2}} - 1$. Allora, $2^{\frac{h^\star}{2}} \leq n + 1$ e $h^\star \leq 2\log(n + 1)$.

C. $h = \mathcal{O}(\log(n))$.

Esercizio 2. **A**. Consideriamo la watershed function $w(n) = n^{\log_b a} = n^0 = 1$ e confrontiamola con la driving function $f(n) = \Theta(1)$. Abbiamo che $f(n) = \Theta(w(n)) = \Theta(w(n) \log^0 n)$, quindi ci troviamo nel secondo caso del teorema Master e la soluzione all'equazione (1) è $T(n) = \Theta(\log(n))$.

B. $T_{\text{Heapify}}(n) = \mathcal{O}(\log(n))$.

C. La radice ha costo cn^2. I nodi dell'i-esimo livello sono 3^i e ognuno ha costo $c\frac{n^2}{16^i}$, quindi il costo dell'i-esimo livello è $3^i c\frac{n^2}{16^i} = cn^2 \left(\frac{3}{16}\right)^i$. L'albero ha altezza $h = \log_4(n)$ e ha $3^h = 3^{\log_4(n)} = n^{\log_4(3)}$ foglie. Allora possiamo scrivere $T(n) = \sum_{i=0}^{n-1} cn^2 \left(\frac{3}{16}\right)^i + dn^{\log_4 3} \leq cn^2 \frac{1}{1-\frac{3}{16}} + dn^{\log_4 3} = cn^2 \frac{16}{13} + dn^{\log_4 3} = \mathcal{O}(n^2)$.