Esame del 14.02.2024

Algoritmi e Laboratorio

Parte B

Esercizio 1. Si consideri l'equazione di ricorrenza

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

A. Si risolva l'equazione (1) al variare del parametro reale $b > 1$, utilizzando il metodo Master.

B. Si stabilisca per quali valori di b la soluzione $T(n)$ all'equazione (1) soddisfa le seguenti condizioni

$$\text{(i.) } T(n) = \mathcal{O}\left(n \log^2(n)\right) \qquad \text{(ii.) } T(n) = \Theta(n) \qquad \text{(iii.) } T(n) = \omega(n \log(n)).$$

C. Si disegni uno sketch dell'albero di ricorrenza associato all'equazione (1) per $b = 3$, indicando l'altezza dell'albero e il numero di foglie.

Esercizio 2. Si consideri il seguente problema computazionale.

Activity Selection Problem

INPUT: un insieme $S = \{a_1, \ldots, a_n\}$ di attività, ognuna con uno *starting-time* e un *finish-time*, ovvero $a_i = [s_i, f_i)$ con $0 \leq s_i < f_i < +\infty$ per ogni $1 \leq n$.

GOAL: selezionare un sottoinsieme di attività $A \subseteq S$ tale che

- le attività in A siano compatibili, cioè, per ogni $a_i, a_j \in A$ distinti si abbia $[s_i, f_i) \cap [s_j, f_j) = \emptyset$;
- A abbia cardinalità massima.

Il problema può essere risolto con un approccio greedy, scegliendo ad ogni iterazione l'attività a_m di S_i con il più piccolo finish-time (ovvero, l'attività che termina prima) e risolvendo il problema per S_{i+1}, che è ottenuto da S_i eliminando a_m e tutte le attività non compatibili con a_m.
Si dimostri che Activity Selection Problem ha la proprietà di scelta greedy.

Soluzioni

Esercizio 1. **A**. La funzione driving e la funzione watershed sono $f(n) = \Theta(n \log(n))$ e $w(n) = n^{\log_b 3}$, rispettivamente.
<u>**Caso** $1 < b < 3$**:**</u> $\log_b 3 > 1$ e quindi per $0 < \varepsilon < \log_b 3 - 1$ si ha che $f(n) = \mathcal{O}\left(n^{\log_b 3 - \varepsilon}\right)$. Allora, per il Teorema Master, $T(n) = \Theta(n^{\log_b 3})$.
<u>**Caso** $b = 3$**:**</u> $\log_b 3 = 1$ e quindi per $k = 1$ si ha che $f(n) = \Theta(n \log n) = \Theta\left(n^{\log_b 3} \log^k n\right)$. Allora, per il Teorema Master, $T(n) = \Theta(n \log^2 n)$.
<u>**Caso** $b > 3$**:**</u> $\log_b 3 < 1$ e quindi per $0 < \varepsilon < 1 - \log_b 3$ si ha che $f(n) = \Omega\left(n^{\log_4 a + \varepsilon}\right)$. Inoltre, è soddisfatta la condizione di regolarità; infatti, per $\frac{3}{b} \leq c < 1$ si ha $3\frac{n}{b} \log \frac{n}{b} \leq c \cdot n \log n$, e per mostrare che un tale c esiste basta osservare che $\frac{3}{b} < 1$ essendo $b > 3$. Allora, per il Teorema Master, $T(n) = \Theta(n \log n)$.

B. (i) $T(n) = \mathcal{O}(n \log^2 n)$ si verifica per $b \geq 3$. Per $1 < b < 3$ si ha invece che $T(n) = \Theta(n^{\log_b 3})$, il quale è di ordine superiore a $\mathcal{O}(n \log n)$ essendo $\log_b 3 > 1$.
(ii) Per $1 < b < 3$ si ha $T(n) = \Theta(n^{\log_b 3})$, con $\log_b 3 > 1$, che è quindi di ordine superiore a $\Theta(n)$. Per $b = 3$ e per $b > 3$, si ha che le relative soluzioni $T(n) = \Theta(n \log^2 n)$ e $T(n) = \Theta(n \log n)$ sono entrambe di ordine superiore a $\Theta(n)$. Quindi, $T(n) = \Theta(n)$ non si verifica per alcun valore di $b > 1$.
(iii) $T(n) = \omega(n \log^2 n)$ per $1 < b \leq 3$; infatti, per $1 < b < 3$, si ha $\lim_{n\to\infty} \frac{n^{\log_b 3}}{n \log n} = \lim_{n\to\infty} \frac{n^{\log_b 3 - 1}}{\log n} = \infty$, e per $b = 3$, si ha $\lim_{n\to\infty} \frac{n \log^2 n}{n \log n} = \lim_{n\to\infty} \log n = \infty$. Invece, per $b > 3$, $\lim_{n\to\infty} \frac{n \log n}{n \log n} = 1$.

C. La radice ha costo $\Theta(n \log n)$. Ogni nodo ha 3 figli. All'i-esimo livello dell'albero ci sono 3^i nodi, ciascuno di costo $\Theta(\frac{n}{3^i} \log \frac{n}{3^i})$. L'altezza dell'albero è $h = \log_3 n$ e ci sono $3^h = 3^{\log_3 n} = n$ foglie.

Esercizio 2. Sia S_k un sottoproblema non vuoto, e sia a_m l'attività di S_k con il minimo finish-time. Per provare che ACTIVITY SELECTION ha la proprietà di scelta greedy, dobbiamo mostrare che esiste una soluzione ottima a S_k che contiene a_m.
Sia A_k una soluzione ottima ad S_k. Supponiamo che a_j sia l'attività con il minimo finish-time in S_k. Se $a_j = a_m$, allora la tesi è vera. Altrimenti, se $a_j \neq a_m$, sappiamo che $f_m \leq f_j$, allora $A'_k = (A_k \setminus \{a_j\}) \cup \{a_m\}$ è una soluzione ottima a S_k. Infatti, $|A'_k| = |A_k|$, e le attività in A'_k sono compatibili perchè le attività in $(A_k \setminus \{a_j\})$ sono compatibili dato che A_k è una soluzione e a_m non si sovrappone a nessuna attività in $(A_k \setminus \{a_j\})$ perchè finisce prima di a_j, che è compatibile con le attività in $(A_k \setminus \{a_j\})$.