Progettazione di Algoritmi **Anno Accademico 2021–2022**

Esercizi sulle Notazioni Asintotiche e l'Analisi di Algoritmi.

Ugo Vaccaro

1. *Esercizio:* In ciascuno dei seguenti casi, indicare se vale che $f(n) = O(g(n))$, oppure se $f(n) = \Omega(g(n))$, oppure valgono entrambe le condizioni (nel cui caso *occorre* indicare che $f(n) = \Theta(g(n))$. Non occorre giustificare le risposte.

   (a) $f(n) = n - \sqrt[3]{n},\ g(n) = 5n$
   (b) $f(n) = 5n \log n,\ g(n) = (n/2)\sqrt[3]{n}$
   (c) $f(n) = 8n + 10\sqrt{n},\ g(n) = n - 10\sqrt{n}$
   (d) $f(n) = n^{3/2} - n,\ g(n) = 9n \log n$
   (e) $f(n) = 7 \log n,\ g(n) = \log n^7$
   (f) $f(n) = \sqrt{\log n},\ g(n) = \log \sqrt{n}$
   (g) $f(n) = 6n^3 + n \log n^2 + \sqrt{n},\ g(n) = n^3 - 4n \log n^2 - 8\sqrt{n}$
   (h) $f(n) = n^{1.01},\ g(n) = n(\log n)^3$
   (i) $f(n) = 2^n,\ g(n) = 3^n$
   (j) $f(n) = 3^{\log_2 n},\ g(n) = 2^{\log_3 n}$
   (k) $f(n) = n2^n,\ g(n) = n!$
   (l) $f(n) = 3^{n+2},\ g(n) = 3^{n-3}$
   (m) $f(n) = n3^n,\ g(n) = 4^n$
   (n) $f(n) = n^2 - n^{3/2},\ g(n) = n^2 - n^{2/3}$
   (o) $f(n) = \log n^4,\ g(n) = \log^2 n$

◇

2. *Esercizio:* Provare le seguenti due affermazioni, **esibendo** le opportune costanti:
   (a) $5n\sqrt{n}\log n^2 + 5n^3 = \Theta(n^3)$
   (b) $(n+2)3^n = O(4^n/n)$.

   Vale che $(n+2)3^n = \Theta(4^n/n)$? Giustificare la risposta.

◇

3. *Esercizio:* Provare le seguenti due affermazioni, **esibendo** le opportune costanti:
   (a) $2n\sqrt[3]{n}\log^2 n^3 + 8n^2 = \Theta(n^2)$
   (b) $n^2 2^n = O(3^n/n^2)$.

   Vale che $n^2 2^n = \Theta(3^n/n^2)$? Giustificare la risposta.

◇

4. *Esercizio:* Provare le seguenti affermazioni, **esibendo** le opportune costanti:
   (a) $7n^2\sqrt[3]{n}\log^3 n^2 + 4n^4 = \Theta(n^4)$
   (b) $10n^3 = O((1.5)^n)$

   Vale che $10n^3 = \Theta((1.5)^n)$? Giustificare la risposta.

$\diamond$

5. *Esercizio:* Date le seguenti funzioni

   $n\log^5 n, \sqrt{n\log n}, \log n^5, n^{\log n}, n^2\log n^2, n^n, n^4, (\log n)^n, 10\sqrt{n}, n\log\sqrt{n}, \log^2 n, n^3\sqrt{n}, 10\log\log n,$

   $3\log n^2, n!$

   ordinarle scrivendole da sinistra a destra in modo tale che la funzione $f(n)$ venga posta a sinistra della funzione $g(n)$ se $f(n) = O(g(n))$. Ad esempio, se le funzioni fossero $3n^2, 3^n, n\log n, \sqrt{n}, \log n$, occorrerebbero scriverle nell'ordine $\log n, \sqrt{n}, n\log n, 3n^2, 3^n$.

$\diamond$

6. *Esercizio:* Date le seguenti funzioni:

   $n-\sqrt{n}, \log(n!/2^n)^4, \log\log n^3, n^2+\log n^2, 3^n+5^n, n\log 2^n, n!, 4n^2+n6\sqrt{n}, 10n+3\log\log n, \log\log n,$
   $5n+2\log n+7\sqrt{n}, \log(n!), 5\cdot 2^n, n\log(n+2)^3, 5n^2+n\log n+n^{3/2}, 4^{\log n}, n\log(n+2)^3, 4^n, n\log\sqrt{n},$

   partizionarle in insiemi disgiunti $A_1, A_2, \ldots$ tali che entrambe le seguenti condizioni valgano
   1. $f(n), g(n) \in A_i \iff f(n) = \Theta(g(n))$
   2. $f(n) \in A_i,\ g(n) \in A_j$, con $i < j \iff f(n) = O(g(n))$ ma $f(n) \neq \Theta(g(n))$.

   Ad esempio, se le funzioni fossero $2n, n+\log n, n^2+\sqrt{n}, 4n^2$, allora

   $$A_1 = \{2n, n+\log n\} \ \text{ e } \ A_2 = \{n^2+\sqrt{n}, 4n^2\}$$

$\diamond$

7. *Esercizio:* Per ciascuna delle seguenti coppie di funzioni f e g, trovare costanti $c \in R_+$ e n_0 per cui $f(n) \leq c\cdot g(n)$, per ogni $n > n_0$.

   (a) $f(n) = 4n^2 + 5n, g(n) = n^2$
   (b) $f(n) = 3\sqrt{n}\log n + 1, g(n) = n + n^2$
   (c) $f(n) = 3n^3 + n + 1 + 1, g(n) = 2n^4$
   (d) $f(n) = n\sqrt{n} + 3n^2, g(n) = n^2$
   (e) $f(n) = 8n + 3, g(n) = n + 7$
   (f) $f(n) = n^2 - n + 1, g(n) = n^2/10$
   (g) $f(n) = 3n + 1, g(n) = (2n^2 + 10)/6$
   (h) $f(n) = 5\sqrt{n}\log n - 1, g(n) = n - \sqrt{n}$

$\diamond$

8. *Esercizio:* Sia $g(n) = n + 2n^3 - 3n^3 + 4n^4$. Dire quali delle seguenti affermazioni sono vere e quali sono false. É necessario provare le affermazioni che si ritengono vere, non è necessario provare le affermazioni che si ritengono false.

   (a) $g(n) = \Omega(n\log n)$
   (b) $g(n) = \Theta(n^5)$
   (c) $g(n) = O(n^{10})$
   (d) $g(n) = \Omega(n^4)$

$\diamond$

9. *Esercizio:* Sia $g(n) = n \log n + 2n^3 - 3n^2$. Dire quali delle seguenti affermazioni sono vere e quali sono false. É necessario provare le affermazioni che si ritengono vere, non è necessario provare le affermazioni che si ritengono false.
   (a) $g(n) = O(n \log n)$
   (b) $g(n) = \Omega(n^2)$
   (c) $g(n) = \Theta(n^3)$
   (d) $g(n) = O(n^4)$

$\diamond$

10. *Esercizio:* Sia $f(n) = 4n^2 + n + 3$. Dire quali delle seguenti affermazioni sono vere e quali sono false, dando la dimostrazione di ogni affermazione.
    (a) $f(n) = \Theta(n^2)$
    (b) $f(n) = \Theta(n\sqrt{n})$
    (c) $f(n) = \Omega(n \log n)$
    (d) $f(n) = O(n \log n)$

$\diamond$

11. *Esercizio:* Sia $f(n) = 4n^3 + n \log n + 5$. Dire quali delle seguenti affermazioni sono vere e quali sono false, dando la dimostrazione di ogni affermazione.
    (a) $f(n) = \Theta(n^3)$
    (b) $f(n) = \Theta(n^2\sqrt{n})$
    (c) $f(n) = \Omega(n^2 \log n)$
    (d) $f(n) = O(n^2 \log n)$

$\diamond$

12. *Esercizio:* Sia $f(n) = 3n \log n + 10\sqrt{n} \log n$. Dire quali delle seguenti affermazioni sono vere e quali sono false, dando la dimostrazione di ogni affermazione.
    (a) $f(n) = O(n^2)$
    (b) $f(n) = \Theta(n \log n)$
    (c) $f(n) = \Omega(n\sqrt{n})$
    (d) $f(n) = O(\sqrt{n} \log^3 n)$

$\diamond$

13. *Esercizio:* Sia $f(n) = 4n\sqrt[2]{n} + 10n \log n + 5$. Dire quali delle seguenti affermazioni sono vere e quali sono false, dando la dimostrazione di ogni affermazione.
    (a) $f(n) = O(n^2)$
    (b) $f(n) = \Theta(n\sqrt[2]{n})$
    (c) $f(n) = \Omega(n \log^2 n)$

(d) $f(n) = O(n \log^2 n)$

$\diamond$

14. *Esercizio:* Date funzioni arbitrarie $f : n \in N \mapsto f(n) \in R_+$ e $g : n \in N \mapsto g(n) \in R_+$ dare le *precise* definizioni di cosa vuol dire che

    (a) $f(n) = O(g(n))$
    (b) $f(n) = \Omega(g(n))$
    (c) $f(n) = \Theta(g(n))$

$\diamond$

15. *Esercizio:* Date funzioni arbitrarie $f : n \in N \mapsto f(n) \in R_+$ e $g : n \in N \mapsto g(n) \in R_+$ provare (utilizzando le definizioni formali di O e Ω) che vale la relazione $f(n) = O(g(n))$ se e solo se vale che $g(n) = \Omega(f(n))$.

$\diamond$

16. *Esercizio:* Date funzioni arbitrarie $f : n \in N \mapsto f(n) \in R_+$, $g : n \in N \mapsto g(n) \in R_+$ e $h : n \in N \mapsto h(n) \in R_+$, provare che se $f(n) = O(g(n))$ e $g(n) = O(h(n))$, allora vale che $f(n) = O(h(n))$, usando la definione di O.

$\diamond$

17. *Esercizio:* Date funzioni arbitrarie $f : n \in N \mapsto f(n) \in R_+$, $g : n \in N \mapsto g(n) \in R_+$ e $h : n \in N \mapsto h(n) \in R_+$, provare che se $f(n) = \Omega(g(n))$ e $g(n) = \Omega(h(n))$, allora vale che $f(n) = \Omega(h(n))$, usando la definione di Ω.

$\diamond$

Ricordiamo il seguente risultato per la risoluzione delle più comuni equazioni di ricorrenza:

Teorema. La soluzione alla ricorrenza

$$T(n) = \begin{cases} d & \text{se } n \leq 1 \\ aT(n/c) + bn^k & \text{altrimenti} \end{cases}$$

per a, c, b, k costanti, è

$$T(n) = \begin{cases} O(n^k) & \text{se } a < c^k \\ O(n^k \log n) & \text{se } a = c^k \\ O(n^{\log_c a}) & \text{se } a > c^k \end{cases}$$

Nel caso in cui non si voglia (o non si possa) applicare il risultato di sopra, gli esercizi che seguono si possono risolvere iterando la equazione di ricorrenza che descrive la complessità di tempo dell'algoritmo e deducendo la soluzione.

1. *Esercizio:* Si consideri il seguente algoritmo:

Algoritmo B(n)
$x = 1$
IF $((n == 1)||(n == 2)$ {
  $x = x + 1$
  } **ELSE** {
    Algoritmo B($n - 2$)
    Algoritmo B($n - 2$)
  }
FOR $(i = 1, i < n + 1, i = i + 1)${
  $x = x + 1$
  }

Si derivi una equazione di ricorrenza che descrive la complessità di **Algoritmo** B(n), e la si risolva, usando il metodo di iterazione.

◇

2. *Esercizio:* Si consideri il seguente algoritmo:

ALGORITMO(n)
IF $(n <= 1)${
RETURN 0
} **ELSE** {
  ALGORITMO($n/2$)
}
  $x = 0; i = 0$
  WHILE $(x \leq n)$ {
    $x = x + 1$
    $i = i + 1$
  }

Si esprima la complessitá di ALGORITMO(n) mediante una equazione di ricorrenza e se ne dia una soluzione in termini della notazione O.

◇

3. *Esercizio:* Si consideri il seguente algoritmo:

Algoritmo S(n)
IF $(n <= 1)$ {
  RETURN 0
  } **ELSE** {
    Algoritmo S($n/3$)
    Algoritmo S($n/3$)
  }
$x = 0$
  WHILE $(x \leq 3n^3)$ {
    $x = x + 3$
  }

Si derivi una equazione di ricorrenza che descrive la complessità di **Algoritmo** S(n), e la si risolva in termini della notazione O.

⋄

4. *Esercizio:* Si consideri il seguente algoritmo:

Algoritmo S(n)
IF ($n <= 1$) {
    RETURN 0
    } **ELSE** {
        Algoritmo S($n/4$)
        Algoritmo S($n/4$)
    }
$x = 0$
WHILE ($x \leq 9\sqrt{n}$) {
        $x = x + 3$
    }

Si derivi una equazione di ricorrenza che descrive la complessità di **Algoritmo** S(n), e la si risolva in termini della notazione O.

⋄

5. *Esercizio:* Si consideri il seguente algoritmo:

Algoritmo S(n)
IF ($n <= 1$) {
    RETURN 0
    } **ELSE** {
    Algoritmo S($n/4$)
    Algoritmo S($n/4$)
    Algoritmo S($n/4$)
    Algoritmo S($n/5$)
    }
$x = 0$
WHILE ($x \leq 3n^2$) {
    $x = x + 3$
    }
$y = 0$
WHILE ($y \leq 3n$) {
    $y = y + 3$
    }

Si derivi una equazione di ricorrenza che descrive la complessità di **Algoritmo** S(n), e la si risolva, in termini della notazione O.