
Algoritmi e Strutture Dati

v0.3.0

Diario delle modifiche

Autore	Versione	Data	Descrizione
Luca De Franceschi	0.3.0	11/06/2014	Inserita spiegazione metodo di sostituzione
Luca De Franceschi	0.2.0	11/06/2014	Inserita teoria su programmazione dinamica
Luca De Franceschi	0.1.0	11/06/2014	Creata struttura del documento

Indice

1	Risoluzione di ricorrenze	4
1.1	Metodo di sostituzione	4
2	Programmazione dinamica	5

1 Risoluzione di ricorrenze

1.1 Metodo di sostituzione

Per capire il metodo di sostituzione proviamo a risolvere il seguente esercizio:

La ricorrenza $T(n) = 4T(n/2) + n^2 \log n$ si può risolvere con il metodo dell'esperto? Giustificare la risposta. Se la risposta è negativa usare il metodo di sostituzione per dimostrare che $T(n) = O(n^2 \log^2 n)$.

Anzitutto vediamo i dati a disposizione:

$$\begin{aligned}a &= 4, b = 2 \\f(n) &= n^2 \log n \\g(n) &= n^{\log_b a} = n^{\log_2 4} = n^2\end{aligned}$$

Calcoliamo ora il limite:

$$\lim_{n \rightarrow +\infty} \frac{n^2 \log n}{n^2} = \infty$$

Da cui deduco che:

$$f(n) = \Omega(n^2)$$

Potrei dunque essere nel caso 3. Devo trovare un

$$\epsilon > 0$$

tale che:

$$\lim_{n \rightarrow +\infty} \frac{n^2 \log n}{n^{2+\epsilon}} \neq 0$$

Ma mi accorgo subito che la cosa è impossibile, in quanto il denominatore, incrementando l'esponente, crescerà molto più velocemente rispetto al numeratore, per cui avrò sempre un valore tendente allo zero. Da questa considerazione deduco che la ricorrenza **non è risolvibile con il metodo dell'esperto**.

Procedo dunque con la sostituzione. Proviamo $T(n) = O(n^2 \log^2 n)$. Assumiamo che per un'opportuna costante $C > 1$ e $\forall x < n$ sia verificata la disuguaglianza $T(x) \leq C(x^2 \log^2 x)$ e dimostriamo che vale anche per n :

$$\begin{aligned}T(n) &= 4T(n/2) + n^2 \log n \leq 4C(n/2)^2 \log^2(n/2) + n^2 \log n \\&= Cn^2(\log n - 1)^2 + n^2 \log n \\&= Cn^2(\log^2 n - 2\log n + 1) + n^2 \log n \\&= Cn^2 \log^2 n - 2Cn^2 \log n + Cn^2 \log n + Cn^2 + n^2 \log n \\&= Cn^2 \log^2 n - (C - 1)n^2 \log n - Cn^2(\log n - 1)\end{aligned}$$

Ora applico una **maggiorazione**:

$$\leq Cn^2 \log^2 n$$

Dunque ho dimostrato che: $T(n) = O(n^2 \log^2 n)$

2 Programmazione dinamica

In maniera del tutto generale la programmazione dinamica può essere descritta nel seguente modo:

1. Identifichiamo dei **sottoproblemi** del problema originario e utilizziamo una *tabella* per memorizzare i risultati intermedi;
2. Inizialmente vanno definiti i **valori iniziali** di alcuni elementi della tabella, corrispondenti a sottoproblemi più semplici;
3. Al generico passo, avanziamo in modo opportuno sulla tabella calcolando il valore della soluzione di un sottoproblema in base alla soluzione di sottoproblemi precedentemente risolti (corrispondenti ad elementi della tabella precedentemente calcolati);
4. Alla fine restituiamo la soluzione del problema originario, che è stato memorizzato in un particolare elemento della tabella.

La programmazione dinamica è usata normalmente per **problemi di ottimizzazione**, il termine “programmazione” si riferisce al metodo tabulare, non alla scrittura di codice.

La programmazione è applicabile con vantaggi se:

- Gode della proprietà di **sottostruttura ottima**: una soluzione si può costruire a partire da soluzioni ottime di sottoproblemi;
- Il numero di sottoproblemi distinti è molto minore del numero di soluzioni possibili tra cui cercare quella ottima, altrimenti c'è la **ripetizione di sottoproblemi**, ovvero se il numero di sottoproblemi distinti è molto minore del numero di soluzioni possibili tra cui cercare quella ottima, allora uno stesso sottoproblema deve comparire molte volte come sottoproblema di altri sottoproblemi.

Ordine di calcolo delle soluzioni dei sottoproblemi

Bottom-up: le soluzioni dei sottoproblemi del problema in esame sono già state calcolate. È il metodo migliore se per il calcolo della soluzione globale servono le soluzioni di tutti i sottoproblemi.

Top-down: è una procedura ricorsiva che dall'alto scende verso il basso. È la soluzione migliore se per il calcolo della soluzione globale servono soltanto alcune delle soluzioni dei sottoproblemi.