Azzolini Riccardo 2019-02-19

Problemi, risorse e complessità

1 Problema

A ogni problema Π si associa una funzione

$$f_\Pi : D_\Pi \to S_\Pi$$

dove:

- D_Π è l'insieme delle istanze di Π
- S_Π è l'insieme delle risposte di Π
- $\forall x \in D_\Pi$, $f_\Pi(x)$ è la soluzione del problema Π relativa a x

2 Algoritmo

A ogni algoritmo ALG si associa una funzione (parziale)

$$\Phi_{\text{ALG}} : \mathit{In}_{\text{ALG}} \to \mathit{Out}_{\text{ALG}} \cup \{\bot\}$$

dove:

- In_{ALG} è l'insieme degli ingressi di ALG
- $\mathit{Out}_{\text{ALG}}$ è l'insieme delle uscite di ALG

ALG **risolve** Π se e solo se

- $\mathit{In}_{\text{ALG}} = D_\Pi$
- $\mathit{Out}_{\text{ALG}} = S_\Pi$
- $\forall x \in D_\Pi \quad f_\Pi(x) = \Phi_{\text{ALG}}(x)$

Si dice allora che ALG è **corretto** per Π.

3 Efficienza

L'**efficienza** di un algoritmo riguarda la quantità di risorse impiegate per la soluzione di un problema.

È necessario fissare

- il modello dell'esecutore
- il tipo delle istruzioni elementari

Le **risorse** sono

tempo: il numero complessivo di istruzioni eseguite

spazio: la memoria occupata (dal programma e dai dati)

L'efficienza viene formalizzata attraverso la nozione di **complessità** (in tempo/spazio).

4 Dimensione dei dati

Per ogni algoritmo ALG con funzione associata Φ_{ALG}, si definisce una funzione

$$W_{\text{ALG}} : In_{\text{ALG}} \to \mathbb{N}$$

che esprime la **dimensione** (il "peso") dei dati in ingresso.

Ad esempio, la dimensione di un intero potrebbe essere il numero di cifre, mentre quella di una matrice potrebbe essere l'ordine o il numero di elementi.

5 Complessità

La **complessità** di un algoritmo ALG si esprime attraverso due funzioni

$$T_{\text{ALG}}, S_{\text{ALG}} : \mathbb{N} \to \mathbb{N}$$

che per ogni $n \in \mathbb{N}$ indicano, rispettivamente, le quantità di tempo e spazio impiegate da ALG per elaborare dati di dimensione n.

Siccome un algoritmo può comportarsi in modo sensibilmente diverso anche per istanze di un problema con la stessa dimensione, la complessità si può misurare in tre casi diversi:

caso peggiore: $T^{w}_{\text{ALG}}(n)$
considera per ogni dimensione l'istanza più *sfavorevole*

caso migliore: $T^b_{\text{ALG}}(n)$
considera per ogni dimensione l'istanza più *favorevole*

in media: $T^a_{\text{ALG}}(n)$
corrisponde alla media dei comportamenti su tutte le istanze aventi ugual dimensione

5.1 Calcolo della complessità in media

In generale, la complessità in media è data dalla formula

$$T^a_{\text{ALG}}(n) = \sum_{i \in In_n} p_i c_i$$

dove:

- In_n è l'insieme delle istanze di dimensione n
- p_i è la probabilità dell'istanza i
- c_i è la quantità di risorse impiegate per elaborare l'istanza i

Raccogliendo i dati di uguale dimensione e costo, si ottiene

$$T^a_{\text{ALG}}(n) = \sum_{k \geq 0} P_k k$$

dove

$$P_k = \sum_{\substack{i \in In_n \\ c_i = k}} p_i$$

Definendo $\sigma_{n,k}$ come il numero di istanze di dimensione n e costo k

$$\sigma_{n,k} = \#\{i \in In_n \mid c_i = k\}$$

e ipotizzando che tutte le istanze in In_n siano *equiprobabili*, si ricava

$$T^a_{\text{ALG}}(n) = \sum_k k \frac{\sigma_{n,k}}{\#In_n} = \frac{1}{\#In_n} \sum_k k \sigma_{n,k}$$

6 Complessità asintotica

La **complessità asintotica** di un algoritmo ALG è il comportamento della funzione $T_{\text{ALG}}(n)$ per grandi valori di n.

Essa è un fattore di primaria importanza nella costruzione di un sistema per la soluzione di un problema. Non è però detto che un algoritmo A_1 con complessità asintotica minore rispetto a un altro algoritmo A_2 (che risolve lo stesso problema) si comporti meglio per ogni dimensione.

6.1 Esempio

$$T_{A_1}(n) = 1000n \qquad T_{A_2}(n) = 2^n$$

Per istanze di dimensione $1 \leq n \leq 13$ conviene utilizzare A_2, mentre per le altre (soprattutto all'aumentare di n) conviene A_1.