

12 - Classificazione dei problemi

I problemi algoritmici possono essere classificati in tre categorie principali:

- Problemi di ricerca
- Problemi di decisione
- Problemi di ottimizzazione

Nello studio dell'informatica la risoluzione di un problema computazionale avviene attraverso due fasi distinte e sequenziali:

- **Concettualizzazione:** È la fase di astrazione in cui si fornisce una specifica formale del problema. Qui si definiscono le caratteristiche strutturali e i vincoli senza preoccuparsi di come il problema verrà risolto.
- **Realizzazione:** È la fase operativa in cui si passa specificando alla scelta della strategia risolutiva, definendo l'algoritmo e implementando il programma.

Specificità di un problema

Per definire formalmente un problema (fase di concettualizzazione), è necessario identificare cinque componenti fondamentali che descrivono le variabili in gioco e le loro relazioni.

1. Lo Spazio di Input

Si stabilisce quali variabili rappresentano i dati di ingresso. L'insieme di tutti i possibili valori che queste variabili possono assumere costituisce lo **spazio di input** del problema.

2. Lo Spazio delle Soluzioni

Si definiscono le variabili che rappresentano i potenziali risultati. L'insieme di tutti i possibili valori per queste variabili costituisce lo **spazio delle soluzioni**.

Questo spazio contiene i candidati a essere soluzione, indipendentemente dal fatto che siano corretti per uno specifico input.

3. La Relazione Caratteristica

È il legame logico imposto dai vincoli del problema. Questa relazione associa gli elementi dello spazio di input agli elementi dello spazio delle soluzioni che sono validi per quell'input. La definizione di questo legame è cruciale per stabilire la correttezza di una soluzione.

4. Lo Spazio di Output

Definisce quali informazioni il processo risolutivo deve restituire. Spesso coincide con lo spazio delle soluzioni, ma può includere simboli aggiuntivi (ad esempio, per indicare che non esiste soluzione).

5. Il Quesito

È la domanda formale a cui l'algoritmo deve rispondere. Stabilisce, dato un input e

basandosi sulla relazione caratteristica, quale elemento dello spazio di output deve essere restituito.

Esempio: Ricerca in un Vettore

Per chiarire le definizioni teoriche, consideriamo il problema di determinare la posizione di un numero intero m all'interno di un vettore V .

Variabili:

- V : un vettore.
- k : un indice (posizione).
- m : un numero intero.

Applicazione della specifica:

- **Spazio di input:** È l'insieme di tutte le possibili coppie formate da un vettore V e un numero intero m .
- **Spazio delle soluzioni:** È l'insieme di tutti i possibili indici (le posizioni) di un vettore.
- **Relazione caratteristica (r_{vett}):** È la regola che associa a una coppia di input (V, m) una posizione k se e solo se l'elemento k -esimo del vettore V è uguale a m .
- **Spazio di output:** Contiene lo spazio delle soluzioni (gli indici) unito a un simbolo speciale che indica l'assenza di soluzioni (nel caso m non sia presente in V).
- **Quesito:** Per ogni configurazione che ammette soluzione, si richiede come output un qualunque elemento dello spazio delle soluzioni che soddisfi la relazione caratteristica.

Analizziamo come agisce la relazione caratteristica r_{vett} su un'istanza specifica.

Consideriamo il seguente input:

- Vettore $V = [7, 5, 3, 7, 2]$
- Valore da ricercare $m = 7$

La relazione caratteristica associa l'input alle soluzioni valide verificando la corrispondenza tra il valore m e gli elementi del vettore.

In questo caso specifico, osserviamo che il valore 7 compare in due posizioni distinte (la prima e la quarta).

Formalmente, la relazione include le seguenti associazioni:

$$\begin{aligned} &<([7, 5, 3, 7, 2], 7), 1> \in r_{vett} \\ &<([7, 5, 3, 7, 2], 7), 4> \in r_{vett} \end{aligned}$$

Questo esempio evidenzia due proprietà fondamentali della relazione caratteristica:

Dimensione Infinita: Poiché esiste un numero teoricamente infinito di possibili vettori generabili, la relazione r_{vett} è un insieme di dimensione infinita.

Molteplicità delle Soluzioni: Per un singolo elemento dello spazio di input (la coppia

vettore-valore), la relazione può associare più valori dello spazio delle soluzioni. Nell'esempio, all'input $(V, 7)$ sono associati sia l'indice 1 che l'indice 4.

La presenza di molteplici soluzioni ammissibili per una singola istanza rende fondamentale la definizione del **Quesito**.

È il quesito, infatti, a stabilire quale sottoinsieme delle soluzioni ammissibili debba essere restituito in output.

Nel contesto di un problema, potremmo essere interessati a diverse tipologie di risultato:

- **Una qualunque soluzione:** È sufficiente restituire uno qualsiasi degli indici validi (es. 1 oppure 4).
- **Tutte le soluzioni:** Si richiede l'elenco completo degli indici in cui compare il valore (es. sia 1 che 4).
- **La "migliore" soluzione:** Si richiede una soluzione che soddisfi un criterio specifico di ottimalità (ad esempio, la prima occorrenza o l'ultima).

Definizione di una specifica di un problema

La specifica di un problema computazionale è definita formalmente come una quintupla:

$$< I, S, R, O, Q >$$

Dove:

- I (Spazio di Input): rappresenta l'insieme di tutti i possibili dati di ingresso accettabili per il problema.
- S (Spazio delle soluzioni): è l'insieme di tutte le possibili strutture che possono fungere da soluzione, indipendentemente dallo specifico input.
- R (Relazione caratteristica): è un sottoinsieme del prodotto cartesiano $I \times S$. Questa relazione definisce quali soluzioni in S sono corrette per un dato input in I .
- O (Spazio di output): è l'insieme dei valori che il problema può restituire al termine dell'elaborato.
- Q (Quesito): è la regola che lega la relazione caratteristica R all'output desiderato. Formalmente, il quesito consente di definire una nuova relazione R_q (sottoinsieme $I \times O$) che stabilisce esattamente cosa restituisce in uscita per ogni input, basandosi sulla validità definita da R .

All'interno di questa struttura, si definisce **istanza** del problema ogni singola occorrenza specifica che si ottiene selezionando un particolare valore all'interno dello spazio di input I .

Mentre il "problema" è la classe generale (es. "ordinare un vettore"), l'"istanza" è il caso concreto da risolvere (es. "ordinare il vettore $[3, 1, 4]$ ").

Problema di ricerca

Questa classe di problemi si concentra sull'individuazione di un risultato valido all'interno dello spazio delle soluzioni.

Data un'istanza specifica del problema, l'interesse è rivolto a:

- Trovare una qualunque delle soluzioni che soddisfano la realizzazione caratteristica. Non è richiesto che sia l'unica o la migliore, è sufficiente che sia ammissibile.
- Nel caso in cui esista alcuna soluzione valida per quell'istanza, acquisire la consapevolezza di tale assenza.

Per formalizzare la possibilità che un problema non abbi soluzione, lo spazio di output viene esteso.

Si introduce un simbolo \perp (si legge: *bottom*), che rappresenta l'assenza di soluzioni.

Problemi di ricerca: definizione

Per comprendere come si applica la definizione formale di un problema di ricerca, analizziamo due esempi classici.

In entrambi i casi, l'obiettivo è trovare una soluzione "ammissibile" (cioè che soddisfa la relazione caratteristica) oppure segnalare che non ne esistono.

Riprendiamo formalmente il problema della ricerca della posizione di un intero in un vettore, definendo come una quintupla:

$$\langle I, S, R, S \cup \{\perp\}, q_{ric} \rangle$$

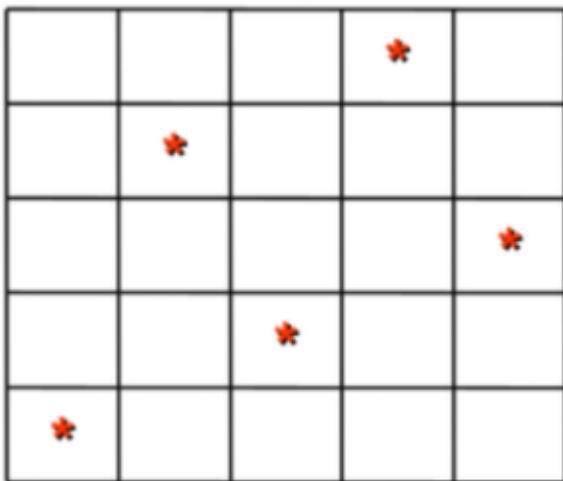
- **Input (I):** È costituito dall'insieme delle coppie $\langle V, m \rangle$, dove V è il vettore e m l'intero cercato.
- **Spazio delle Soluzioni (S):** È l'insieme degli interi positivi, che rappresentano i possibili indici del vettore.
- **Relazione Caratteristica (R):** È la relazione r_{vett} che lega l'indice al valore contenuto in quella posizione.
- **Obiettivo:** Per ogni istanza, siamo interessati a calcolare una delle soluzioni valide. Se l'elemento m non è presente nel vettore V , il sistema deve restituire il simbolo \perp .

In questo contesto, "trovare una soluzione ammissibile" significa individuare un indice k tale che $V[k] = m$.

Esempio: problema della N regina

Dato un numero intero positivo N , si deve determinare un posizionamento di N regine su una scacchiera di dimensioni $N \times N$ in modo tale che nessuna regina possa "minacciare" un'altra regina (secondo le regole degli scacchi, una regina non deve trovarsi sulla stessa

riga, colonna o diagonale di un'altra).



Il problema è definito dalla quintupla $\langle N^+, D, R, D \cup \{\perp\}, q_{ric} \rangle$, dove:

- N^+ : l'insieme dei numeri positivi. L'input è semplicemente un numero N (es. $N = 5$).
- D : l'insieme di tutte le possibili soluzioni possibili di regine su scacchiere di qualsiasi dimensione. Qui si assume che il numero di regine sia uguale alla dimensione della scacchiera.
- R : contiene tutte e sole le coppie $\langle x, y \rangle$ dove:
 - x è l'intero positivo in input ($x \in N^+$)
 - y è una specifica disposizione di x regine su una scacchiera $x \times x$ che rispetta il vincolo: nessuna regina minaccia le altre.

Risolvere questo problema significa trovare una configurazione y all'interno di D che dia legata all'input N della relazione R . Se per un dato N non esiste alcuna configurazione sicura, l'output sarà \perp .

Esempio 2: ordinamento di un vettore in maniera crescente

Un problema di ricerca nell'ordinare gli elementi di un vettore in modo crescente. Anche in questo caso, è fondamentale distinguere tra la specifica del problema e l'algoritmo che lo risolve.

Il problema dell'ordinamento è definito definendo lo spazio delle soluzioni come manipolazione degli indici.

- I : è l'insieme di tutti i possibili vettori interni, di qualunque dimensione.
- S : è l'insieme di tutte le possibili sequenze di indici. Per un vettore di dimensione n , lo spazio delle soluzioni corrisponde a tutte le possibili permutazioni degli interi da 1 a n .
- R : la realizzazione include le coppie $\langle V, W \rangle$ dove V è il vettore di input e W è il vettore degli indici soluzione che soddisfano la condizione di ordinamento. Formalmente, se $V = [v_1, \dots, v_n]$ e $W = [w_1, \dots, w_n]$, la coppia è valida se, scorrendo V secondo l'ordine dettato da W , gli elementi risultano crescenti:

$$V[W[q+1]] \geq V[W[q]]$$

per ogni posizione q valida.

Considerando il vettore di input $V = [7, 1, 3, 4]$. La soluzione non è il vettore ordinato $[1, 3, 4, 7]$, bensì il vettore degli indici (le loro posizioni) $W = [2, 3, 4, 1]$. Infatti:

- L'elemento in posizione 2 è 1.
- L'elemento in posizione 3 è 3.
- L'elemento in posizione 4 è 4.
- L'elemento in posizione 1 è 7.

Questa logica è indipendente dal tipo di dato; si applica identicamente a vettori di caratteri o altri tipi ordinabili (es. $V = [a, z, p, b] \rightarrow W = [1, 4, 3, 2]$).

Problemi di decisione

In un problema di decisione, l'obiettivo è stabilire se il dato in ingresso soddisfa o meno una certa proprietà. Di conseguenza, lo spazio di output è ristretto ai soli valori di verità:

$$O = \{\text{true}, \text{false}\}$$

Un problema di decisione P è specificato dalla quintupla:

$$\langle I, S, R, \{\text{true}, \text{false}\}, q_{dec} \rangle$$

A differenza dei problemi di ricerca, dove si restituisce un elemento di S , qui il quesito definisce una funzione che mappa la relazione caratteristica in un valore booleano. Per ogni istanza i del problema, l'output sarà:

- True: se esiste almeno una soluzione s nello spazio S tale che la coppia $\langle i, s \rangle$ appartenga alla relazione caratteristica R (ovvero esiste una sola soluzione ammissibile).
- False: altrimenti se non esiste alcuna soluzione che soddisfi la relazione con l'input i .

Strategia di risoluzione per i problemi decisionali

La definizione di un problema di decisione P pone il quesito in termini di vero o falso.

Il metodo più diretto consiste nel trasformare il problema di decisione P nel suo corrispettivo di ricerca P' .

La logica risolutiva procede:

- Si esegue l'algoritmo risolutivo per il problema di ricerca P' .
- Si analizza l'output prodotto:
 - Se l'algoritmo restituisce il simbolo \perp , allora la risposta al problema di decisione P è false.
 - Se l'algoritmo restituisce una qualunque soluzione ammissibile s , allora la risposta al problema di decisione P è true.

Questo approccio è detto "**costruttivo**" perché la dimostrazione dell'esistenza della soluzione avviene attraverso la sua effettiva costruzione o individuazione.

Esiste una seconda via che non richiede la risoluzione del problema di ricerca associato. In questo caso, si mira a trovare una soluzione a P senza risolvere il corrispettivo problema di ricerca P' .

Problema della partizione

Sono dati k numeri interi positivi n_1, \dots, n_k la cui somma è $2m$ (per un certo intero m).

Decidere se i k numeri possano essere ripartiti in due gruppi in modo che la somma dei componenti di ogni gruppo sia m .

Dunque se:

$$n_{a_1}, \dots, n_{a_p} \in n_{b_1}, \dots, n_{b_q}$$

Sono i due gruppi di numeri (cioè $p + q = k$), deve essere:

$$\sum_{i=1}^p n_{a_i} = \sum_{j=1}^q n_{b_j} = m$$

Passando alla specifica formale di questo problema, possiamo definire le componenti della quintupla come segue:

- I : è costituito dall'insieme di tutti i possibili insiemi di interi che soddisfano una condizione necessaria: la loro somma totale deve essere un numero pari.
- S : è l'insieme di tutte le possibili coppie di interi.
- R : è l'insieme di tutte e sole le coppie $\langle x, y \rangle$ dove x appartiene allo spazio di input e y allo spazio delle soluzioni. Nello specifico, se la soluzione y è una coppia di insiemi $\langle y_1, y_2 \rangle$, affinché la relazione sia soddisfatta devono valere contemporaneamente tre condizioni:
 1. I due insiemi devono essere disgiunti: $y_1 \cap y_2 = \emptyset$.
 2. L'unione dei due insiemi deve ricostruire l'input originale: $y_1 \cup y_2 = x$.
 3. Le somme degli elementi dei due insiemi devono coincidere (o essere uguali): $\sum(y_1) = \sum(y_2)$.

Per esempio:

Consideriamo un insieme di input rappresentato dal vettore:

$$[1, 3, 2, 7, 5]$$

Questo insieme può essere correttamente partizionato nei due sottogruppi:

- $[2, 7]$
- $[1, 5, 3]$

Verifichiamo la correttezza: la somma del primo gruppo è 9 ($2+7$) e la somma del

secondo gruppo è anch'essa 9 (1+5+3).

Problema di ottimizzazione

Nei problemi di ottimizzazione alle soluzioni ammissibili è associata una misura o costo/valore.

L'obiettivo non è trovare una soluzione qualsiasi, ma la migliore possibile secondo criteri di preferenza stabiliti (come anche massimizzare un profitto o minimizzare un costo).

Il quesito Q diventa una tripla $q_{ott}(M, m, \subseteq)$, dove m è la **funzione obiettivo** che calcola il valore della soluzione e \subseteq è l'ordinamento che stabilisce cosa significa "migliore" su M .

Esempio: livello dell'albero a peso massimo

E' dato in input un albero i cui nodi sono etichettati con valori interi non negativi. Si definisce "peso di un livello" la somma delle etichette di tutti i nodi che si trovano a quel livello.

L'obiettivo è individuare quale livello dell'albero possiede il peso massimo.

- **Spazio di input(I)**: è l'insieme di tutti i possibili alberi con nodi etichettati da interi non negativi.
- **Spazio delle soluzioni(R)**: collega un albero T a un intero n è un livello effettivamente esistente in quell'albero. Formalmente, se l'albero ha profondità p , deve valere $n \leq p$.
- **Spazio di output**: è $S \cup \{\perp\}$, per gestire eventuali casi degeneri.
- **Quesito di ottimizzazione**: è definito dalla regola $q_{ott}(N, m; \geq)$, composta da:
 1. Insieme di misura (N) i valori di peso sono numeri naturali.
 2. Funzione obiettivo (m) è una funzione $m(T, n)$ che, dato l'albero T e il livello n , calcola la somma delle etichette dei nodi presenti a quel livello.
 3. Criterio (\geq) cerchiamo il massimo valore possibile generato dalla funzione m .

Esempio del problema dello zaino

Immaginiamo un ladro che si introduce in un negozio. Si trova di fronte a n articoli. Ogni articolo i -esimo possiede due caratteristiche:

- Un valore (profitto) P_i (in euro).
- Un peso (costo) C_i (in kg). Per semplicità, assumiamo che tutti questi valori siano interi.

Il ladro ha uno zaino che può supportare un peso massimo B . L'obiettivo è decidere quali oggetti prendere per massimizzare il valore totale della raffurtiva, rispettando il vincolo di capacità dello zaino.

Per formalizzare il problema occorre considerare $2n + 1$ interi positivi in input:

- P_1, \dots, P_n : i profitti degli oggetti.
- C_1, \dots, C_n : i costi (pesi) degli oggetti.
- B : il peso massimo.

Dobbiamo trovare n valori interi X_1, \dots, X_n che fungono da variabili decisionali, dove:

- $X_i \in \{0, 1\}$: se $X_i = 1$ l'oggetto viene preso, se $X_i = 0$ viene lasciato.
I vincoli e l'obiettivo sono:
- Funzione obiettivo: $\sum_{i=1}^n P_i X_i$ la somma dei profitti degli oggetti scelti.
- Vincolo di capacità: $\sum_{i=1}^n C_i X_i \leq B$ il peso totale non deve essere superato.

Utilizziamo lo schema formale $\langle I, S, R, O, Q \rangle$, definiamo il problema come segue:

1. Spazio di input (I): è l'insieme delle coppie (n, q) dove n è un intero positivo (numero di oggetti) e q è una sequenza di $2n + 1$ interi positivi. L'elemento tipico di questo spazio è strutturato come:

$$(n, (P_1, \dots, P_n, C_1, \dots, C_n, B))$$

2. Spazio delle soluzioni (S): è l'insieme delle sequenze finite binarie $\langle X_1, \dots, X_n \rangle$ con $n \in N^+$ e $X_i \in \{0, 1\}$. Questo rappresenta tutte le possibili combinazioni di "prendere o lasciare" gli oggetti, indipendenti dal peso.
3. Relazione caratteristica (R): è l'insieme delle coppie (input, soluzione) tali che la soluzione rispetti il vincolo di peso imposto dall'input. Formalmente, la coppia è valida se:

$$\sum_{i=1}^n C_i X_i \leq B$$

4. Quesito (Q): il quesito è di ottimizzazione: $q_{ott}(N, m, \geq)$. La funzione obiettivo m da massimizzare è definito come:

$$m = \sum_{i=1}^n P_i X_i$$