

## **Algoritmo**

In altri termini, l'algoritmo può essere definito come **la descrizione** di un procedimento che ha le seguenti proprietà: deve essere esplicita e non ambigua per l'interlocutore cui è destinata e il procedimento attivato di volta in volta seguendo il testo della descrizione deve terminare in un tempo finito. Il concetto di algoritmo presuppone quindi sempre la presenza di un linguaggio usato per la descrizione e di due interlocutori (chi produce la descrizione e chi sa eseguire tutte le azioni previste) che lo sanno usare.

## **Calcolabilità**

La teoria della calcolabilità si occupa dei criteri che consentono di definire la effettiva risolubilità dei problemi. Ne consegue che un problema si può risolvere in modo effettivo se e solo se esiste un algoritmo per trattare qualsiasi istanza di quel problema. La importanza pratica di questa affermazione sta nel fatto che con i linguaggi di programmazione per computer è possibile descrivere qualsiasi algoritmo.

## **Complessità**

La teoria della complessità definisce una condizione necessaria per applicare una strategia costruttiva per risolvere un problema: l'algoritmo individuato deve essere computazionalmente trattabile. Quando la soluzione di un problema dipende da un parametro  $n$ , detto anche dimensione del problema generico (ciò che si verifica per esempio quando l'input è un insieme di  $n$  elementi), il problema è trattabile se le risorse (tempo di calcolo e spazio di memoria) crescono in modo polinomiale (e non esponenziale) rispetto alla sua dimensione.

## **Computer**

Nel contesto del problem solving, il computer è un interlocutore capace di eseguire elaborazioni di informazioni descritte utilizzando un linguaggio di programmazione.

### **Correttezza**

La teoria della correttezza si occupa dei problemi e delle metodologie per dimostrare che il comportamento di un programma è coerente con lo scopo per cui è stato scritto. Un programma corretto risolve in modo esatto qualsiasi istanza del problema per cui è stato scritto. Nella pratica è difficile dimostrare la correttezza di un programma; per questo è sempre necessario sottoporre il programma a verifiche (parziali) di correttezza utilizzando una scelta appropriata di casi di prova.

### **Input**

L'input di un problema è rappresentato dall'insieme dei dati noti che definiscono le singole istanze di quel problema che devono essere risolte: per questo motivo spesso si usa la locuzione "dati di input".

### **Intelligenza Artificiale**

Con questo termine, nato nel 1956 durante un convegno al quale parteciparono i padri fondatori di questa disciplina (tra i quali McCarthy, Minsky, Simon e Newell), si fa riferimento al settore dell'Informatica che si occupa della realizzazione di sistemi artificiali capaci di comportamenti che un osservatore comune attribuirebbe all'intelligenza umana; i sistemi artificiali "intelligenti" sono in genere abbastanza complessi e sanno apprendere e adattarsi (con successo) all'ambiente.

### **Istanza di un problema**

Dato un problema generico (per esempio calcolare l'area di un rettangolo), una istanza specifica è rappresentata da un particolare problema di cui si chiede la soluzione (per esempio trovare l'area di un rettangolo con base uguale a 3 cm e altezza uguale a 5 cm). Quindi una istanza specifica di

un problema viene definita quando sono noti tutti i dati di input.

### ***Linguaggio di programmazione***

E' un linguaggio artificiale con capacità espressive definite nell'ambito della teoria della calcolabilità (il cui rispetto garantisce che con linguaggi di questo tipo è possibile descrivere qualsiasi algoritmo). I linguaggi di programmazione vengono usati per descrivere (facilmente) procedimenti di elaborazione che devono essere eseguiti da un computer. La nascita dei linguaggi di programmazione è successiva alla comparsa dei computer; la loro motivazione principale è stata quella di semplificare l'attività di scrivere programmi per computer senza richiedere la conoscenza delle particolarità hardware del computer.

### ***Lista***

La lista è una particolare struttura che consente di rappresentare tutti gli elementi di una collezione di dati (per esempio i nomi di un elenco, gli elementi di un insieme o le componenti di un vettore); viene usata da particolari linguaggi di programmazione per descrivere i dati coinvolti in elaborazioni descritte da un programma.

### ***Output***

In opposizione a input (che rappresenta i dati di un problema), l'output rappresenta il risultato della soluzione di una particolare istanza di un problema.

### ***Problema di decisione***

I problemi di decisione sono quei problemi il cui output è rappresentato semplicemente da un "si" oppure da un "no". In genere, i problemi di decisione consistono nel verificare se i dati di input soddisfano opportuni criteri.

### ***Problema generico***

Un problema generico (in contrapposizione a istanza specifica) è un problema formulato facendo riferimento solo alle entità coinvolte, senza specificare alcun dato di input (per esempio calcolare l'area di un rettangolo conoscendone le misure della base e dell'altezza). La soluzione di un problema generico si ottiene esibendo un procedimento di soluzione;

mentre la soluzione di una istanza specifica, assegnati i dati di input, si ottiene producendo i valori dell'output.

### **Programma**

Un programma è la descrizione di un algoritmo eseguita utilizzando un linguaggio di programmazione. In genere, un algoritmo viene descritto come programma quando il processo di elaborazione deve essere svolto da un computer.

### **Programmazione**

La programmazione, in Informatica, è l'attività che viene svolta per ottenere programmi per computer; essa è parte essenziale del problem solving quando si assume che l'interlocutore/esecutore sia un computer (ovvero quando si vuole che il procedimento sia descritto in modo esplicito e non ambiguo e quindi sia eseguibile in modo meccanico dall'esecutore).

### **Scomposizione**

La scomposizione di un problema in sottoproblemi è l'attività centrale del processo di problem solving; mediante la scomposizione, la soluzione di un problema (complesso) viene ricondotta alla soluzione di un elenco di sottoproblemi (più semplici). Se la scomposizione viene descritta utilizzando un linguaggio di programmazione (per esempio il Prolog), essa è interpretabile come programma eseguibile da un computer.

### **Stringa**

Una stringa è una qualsiasi sequenza costruibile con i caratteri disponibili; con i linguaggi di programmazione, i caratteri disponibili sono quelli presenti sulla tastiera collegata al computer. A seconda del significato da attribuire alle stringhe, possono esserci particolari limitazioni: per esempio alcuni tipi di stringhe possono essere costruite solo coi caratteri alfabetici e con le cifre numeriche; altre devono iniziare con una lettera minuscola (o maiuscola) dell'alfabeto. I vincoli sull'uso dei caratteri per costruire i vari tipi di stringhe sono definiti dalla sintassi del linguaggio di programmazione che si deve usare.

### **Variable**

La variabile in Informatica, è la struttura sintattica usata nei programmi per rappresentare le entità coinvolte nelle elaborazioni. Le variabili di input vengono istanziate (cioè assumono valori) all'inizio del processo di elaborazione, quelle di output durante questo processo. Lo scopo di un programma è quello di trovare i valori delle variabili di output che corrispondono a (o sono desumibili da) i dati di input.