

## 6 - Linguaggi di programmazione

Sono appartenenti alla classe dei linguaggi di alto livello e ispirati alla struttura fisica del calcolatore (architettura di Von Neumann, istruzioni di macchina e memorizzazione di valori in celle di memoria)

Un programma quindi è un insieme di comandi "imperativi", la loro computazione consiste in una sequenza di passi che modificano lo stato, mettendo a disposizione del programmatore un insieme di dati primitivi e di meccanismi di composizione che consentono di organizzare in modo agevole i dati relativi al problema

### Struttura

La struttura del programma consiste in

- Parte di dichiarazione in cui si dichiarano tutte le variabili del programma insieme al tipo
- Parte di istruzioni che descrive l'algoritmo risolutivo utilizzato mediante le istruzioni  
Le istruzioni poi si dividono in
  - Lettura e scrittura
  - Assegnamento (un astrazione della cella di memoria)
  - Controllo

### Comunicazione dell'algoritmo all'elaboratore

I linguaggi devono essere non ambigui e sequenziali per essere comprensibili alla macchina, con requisiti di descrizione:

- **Univocità** (non dare ambito a interpretazioni errate)
- **Completa** (prevede tutte le azioni necessarie)
- **Ripetibile** (garantire un buon risultato se eseguita da più esecutori con medesime caratteristiche)

### Programma

Il programma non è altro che la traduzione in un linguaggio comprensibile alla macchina, della procedura di soluzione, con indicazione sui dati di ingresso e di uscita.

Comunica al calcolatore le istruzioni operative (quali dati di ingresso deve trattare, come deve operare su questi dati, quali dati deve dare come risultato) ed è una procedura eseguibile sul calcolatore che rappresenta una soluzione al problema (corrisponde alla tripla: Dati, Algoritmo, Risultati)

### Dati

I dati sono entità su cui un problema lavora, possono essere variabili o costanti a seconda del tipo.

I dati su cui lavora un programma sono rappresentati come sequenze di bit, nei linguaggi ad alto livello, il programmatore può ignorare i dettagli della rappresentazione fisica dei dati, concentrandosi sui tipi di dato.

## Istruzioni

Le istruzioni si dividono in dichiarative, che definiscono come rappresentare le entità del problema, e operative, che lavorano su queste rappresentazioni.

### Istruzioni dichiarative

Le istruzioni dichiarative definiscono le aree di memoria in cui sono conservati i dati e che fa riferimento un algoritmo, predispongono le posizioni di memoria da utilizzare e associano un nome (identificatore) a ciascuna di esse, determinano anche il tipo di dati che possono essere memorizzati (cioè insieme di valori permessi e operazioni applicabili)

## Variabili

Una variabile è un nome simbolico che denota una locazione di memoria del computer contraddistinta da uno specifico indirizzo e tramite essa il valore su cui applicare le istruzioni del programma. Ad ogni esecuzione di un processo si usano i valori associati ai nomi simbolici che compaiono nell'algoritmo che lo descrive

Ogni variabile è caratterizzata da un nome, un valore e un tipo. Le variabili assumono valore tramite istruzioni di ingresso o assegnamento.

Il nome della variabile denota una coppia:

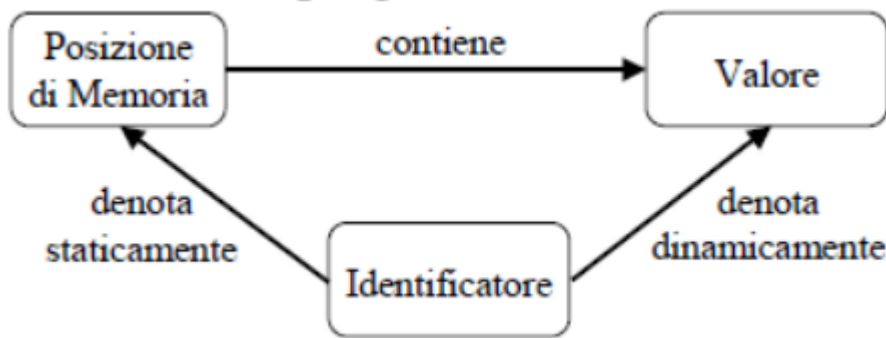
- Posizione di memoria
- Quantità in essa contenuta (Una limitazione nel rappresentare dati di tipo numerico alfanumerico viene dalle dimensioni limitate della memoria)

Una variabile assume valore quando nella locazione ad essa riservata viene memorizzato il valore, ciò avviene tramite istruzioni di ingresso o di assegnamento

## Legame degli identificatori

Il legame tra identificatore e posizione di memoria è statico e non cambia durante l'esecuzione del programma, mentre il legame tra identificatore e valore è dinamico e può

variare.



## Identificatori in C

### ✓ Esempi

1)  $X = 10$  Assegnamento del valore 10

2)  $X = X + 1$  Incremento unitario

dopo 1) X varrà 10

dopo 2) X varrà 11

3)  $Y = X$  Copia

- nella locazione legata ad Y viene copiato il valore contenuto nella locazione legata ad X (sovrascrittura, si perde il valore di Y precedente all'assegnamento)

## Costanti

Le costanti sono dati il cui valore non varia durante l'esecuzione del programma. Sono accessibili solo in lettura, garantendo che non possano essere modificate accidentalmente.

## Tipi di istruzioni

Le istruzioni si dividono in istruzioni di ingresso e uscita, istruzioni di assegnamento e istruzioni di controllo. Le istruzioni di ingresso e uscita permettono la comunicazione con l'utente, mentre le istruzioni di assegnamento e controllo gestiscono la logica del programma.

### Istruzioni di ingresso e uscita

Le istruzioni di ingresso permettono di acquisire dati dall'esterno, inserendoli in opportune variabili del programma. Le istruzioni di uscita consentono di notificare all'utente i risultati del programma, copiando i dati su un supporto esterno.

### Istruzioni di assegnamento

L'istruzione di assegnamento produce un nuovo valore e lo assegna a una variabile. L'espressione viene valutata e il valore ottenuto viene memorizzato nella variabile. La variabile che appare alla sinistra del simbolo di assegnamento assume il valore di ciò che appare alla destra.

$$\begin{array}{l} x \longleftarrow y \neq y \longleftarrow x \\ x \longleftarrow x + 1 \end{array}$$

## Linguaggio

I linguaggi naturali sono ambigui, con significati che dipendono dal contesto in cui sono inseriti. I linguaggi artificiali, come i linguaggi di programmazione, devono essere precisi (regole e parole certe, non cambiano con l'uso) e non ambigui (non dipendenti dal contesto, la macchina non ha possibilità di più interpretazioni) per essere comprensibili alla macchina.

La sintassi di un linguaggio riguarda la struttura delle frasi, mentre la semantica si occupa del significato delle frasi. Entrambi gli aspetti sono fondamentali per la comprensione e l'uso corretto di un linguaggio di programmazione.

Un linguaggio di programmazione è una notazione comprensibile al calcolatore per rappresentare un algoritmo.

## Messaggio

Un messaggio è una sequenza di frasi espresse in un linguaggio, analizzabile dal punto di vista sintattico e semantico. La sintassi verifica la forma linguistica, mentre la semantica individua il significato associato.

Un programma è un messaggio di comunicazione tra l'uomo e la macchina, composto da frasi costruite secondo regole rigide per eliminare ambiguità nell'interpretazione dei comandi.

Per passare da un linguaggio ad alto livello a un linguaggio macchina è necessario un traduttore. Esistono due tipi di traduzione: interpretazione e compilazione.

- **Interpretazione:** Gli interpreti traducono ed eseguono una riga del programma alla volta. La traduzione procede passo passo con l'esecuzione, traducendo ogni istruzione tante volte quante viene eseguita. (Un esempio è SQL)
- **Compilazione:** Gli interpreti traducono ed eseguono una riga del programma alla volta. La traduzione procede passo passo con l'esecuzione, traducendo ogni istruzione tante volte quante viene eseguita. (Un esempio è C)
- Un linguaggio ibrido tra loro due è Java