

LINGUAGGI

SISTEMI INFORMATIVI E DBMS

CORSO DI LAUREA MAGISTRALE IN
MANAGEMENT E MONITORAGGIO DEL TURISMO SOSTENIBILE



PROF. ANDREA PINNA

AA 2019/2020

LINGUAGGI

Linguaggi naturali: strumento di comunicazione usato dai membri di una stessa comunità, evoluti nel tempo insieme all'uomo.

Linguaggi formali: sviluppati per specifiche applicazioni. Si basano su una convenzione.
Esempio: linguaggio matematico

$$\forall x \in X : x > 0$$



LINGUAGGI

Linguaggi di programmazione: sono linguaggi formali concepiti per esprimere le **istruzioni** di un programma da far eseguire ad un calcolatore elettronico.

Programma: sequenza di istruzioni scritte in un linguaggio di programmazione.



SEMANTICA

Semantica: esprime il **significato** dei costrutti del linguaggio. Può essere informale (ad esempio un manuale) o formale (espressa tramite formalismo matematico).

Due diverse tipologie di semantica:

Operazionale: descrive in modo preciso il funzionamento dei costrutti del linguaggio.

Dichiarativa: esprime il significato dei costrutti del linguaggio dicendo cosa esprimono ma non come funzionano



SINTASSI

La sintassi è l'insieme di regole per la creazione di "frasi" in un dato linguaggio. La sintassi è composta da due elementi:

Token ammissibili: singoli elementi che costituiscono una frase o una istruzione.

Esempio: sono ammessi i simboli 1 , x , $=$

Strutture di frase: regole grammaticali che dicono come comporre i token per realizzare una frase o un'istruzione.

Esempio: l'istruzione $x = 1$ è detta **assegnamento**

Il processo che permette date le regole di comprendere la struttura di una frase è chiamato Parsing.



LIVELLO DI ATRAZIONE

Un linguaggio di programmazione può essere classificato in base al suo livello di astrazione.

I linguaggi a **basso livello** di astrazione (o linguaggi di basso livello) sono concepiti per essere direttamente tradotti in linguaggio macchina dal calcolatore.

Esempio: linguaggio assembly

$a = (b+c) - (d+e)$	→	<code>add \$a0, \$a0, \$a1</code>
		<code>add \$v0, \$a2, \$a3</code>
		<code>sub \$v0, \$a0, \$v0</code>



LIVELLO DI ATRAZIONE

I linguaggi ad **alto livello** di astrazione (o linguaggi di alto livello) sono maggiormente comprensibili dall'uomo. Prima di poter essere eseguiti dal calcolatore necessitano di una **traduzione** che richiede tempo di elaborazione.

Esempi; c, c++, **Python**, solidity, javascript e tanti altri

Generalmente un programma scritto in un linguaggio ad alto livello si chiama **codice sorgente**.



COMPILATO VS INTERPRETATO

Il codice sorgente scritto in un linguaggio ad alto livello può essere tradotto in linguaggio macchina in due modi:

- Compilazione
- Interpretazione



LINGUAGGI COMPILATI

Compilazione: il codice sorgente viene trasformato in un file “**eseguibile**” scritto in linguaggio macchina compatibile con il sistema operativo in uso.

Esempi: c. c++

Vantaggi: il programma è ottimizzato e risulta più veloce.

Svantaggi: i file eseguibili funzionano soltanto sui sistemi operativi compatibili. Ad esempio, un programma scritto in linguaggio c e compilato per Linux non gira su Windows e viceversa.



LINGUAGGI INTERPRETATI

Interpretazione: il codice sorgente viene esaminato dall'interprete istruzione per istruzione, tradotto in linguaggio macchina ed eseguito dal calcolatore.

Esempio: Python

Vantaggi: i codici sorgente sono direttamente eseguibili e funzionano su tutti i sistemi operativi in cui è installato un interprete del linguaggio.

Svantaggi: il programma non è ottimizzato e risulta più lento.

