

Lab. Programmazione (CdL Informatica)
&
Informatica (CdL Matematica)
a.a. 2022-23

Monica Nesi

Università degli Studi dell'Aquila

11 Ottobre 2022

Linguaggio di Programmazione

Definisce un insieme di *simboli* e *parole* ed un insieme di *regole* per combinarli (lessico + sintassi) per ottenere *frasi* del linguaggio (programmi) corretti sintatticamente.

Ogni linguaggio di programmazione definisce una *macchina astratta*.

Primi linguaggi di programmazione (anni '50 del secolo scorso):

- FORTRAN (calcolo numerico, 1954)
- COBOL (applicazioni gestionali, 1959)
- LISP (linguaggio funzionale, John McCarthy, 1958).

Linguaggi di Programmazione: alcune classificazioni

I linguaggi di programmazione possono essere classificati in modi diversi, secondo criteri o parametri diversi.

Una classificazione si basa sul concetto di *livello*:

- ▶ linguaggi ad alto livello;
- ▶ linguaggi assembler (livello intermedio);
- ▶ linguaggi a basso livello (linguaggi macchina).

Linguaggi di Programmazione: alcune classificazioni (cont.)

Sia L un linguaggio di programmazione. L si dice:

- ▶ *compilato* se esiste un programma (detto *compilatore* per L) che traduce un programma scritto in L (*codice sorgente*) in un programma scritto in linguaggio macchina (*codice oggetto* o *codice macchina*), direttamente eseguibile su un calcolatore. Il compilatore esegue analisi lessicale, analisi sintattica e controllo dei tipi (typechecking).
Esempi: ADA, Algol, C, Pascal, LISP, FORTRAN, COBOL, Haskell, ML, ...
- ▶ *interpretato* se esiste un programma (detto *interprete* per L) che esegue direttamente un programma scritto in L , mentre fa controlli di correttezza.
Esempi: LISP, JavaScript, MATLAB, Python, SmallTalk, Java, ...

Paradigmi di programmazione

I linguaggi ad alto livello si possono distinguere in base al *paradigma* di programmazione. I principali sono:

- ▶ *imperativo*: Algol60 + AlgolW, Pascal, C, ADA (concorrente),
...
- ▶ *funzionale*: LISP, ML, CAML, SML, Haskell, Miranda, ...
- ▶ *logico*: PROLOG
- ▶ *orientato agli oggetti*: SmallTalk (puro), Simula, C++, Java,
...
- ▶ altri paradigmi (visuale, scripting, web, ...)

Linguaggio Java

Nato nei primi anni '90 ad opera della Sun Microsystems (prima versione ufficiale nel 1996)

Linguaggio orientato agli oggetti (*object-oriented*)

Un programma in Java è una collezione di *oggetti* che soddisfano certi requisiti e che interagiscono tramite delle funzionalità rese pubbliche (approccio basato su *tipi di dato astratti*)

Concetti di *classi*, *oggetti* e *metodi*.

Portabilità: Java è nato per essere *portabile* ed eseguibile su macchine ed architetture diverse (*linguaggio indipendente dalla piattaforma*) ed orientato alle applicazioni di rete.

Bytecode e JVM

Java: linguaggio *interpretato* ma con una *leggera* compilazione.

Dato il codice sorgente Java, la compilazione genera il codice detto *Bytecode* (un programma non legato alla macchina), che poi viene eseguito dall'interprete di Java.

La Java Virtual Machine (JVM) è l'interprete di Java, costruito in modo da garantire un certo grado di *sicurezza* (l'interprete effettua un certo tipo di controlli mentre esegue il programma).

Java è un linguaggio *fortemente tipato* (strongly typed).

È un linguaggio *robusto*: meccanismi per la gestione della memoria (garbage collector), gestione delle eccezioni, ...

Ereditarietà e sottotipi (sottoclasse)

Ricca libreria di oggetti a disposizione