

Complementi di Programmazione

Introduzione ai Linguaggi Dinamici

CdL Informatica - Università degli studi di Modena e Reggio Emilia
AA 2023/2024

Filippo Muzzini

Cenni storici

Gli sviluppi nei linguaggi di programmazione rispecchiano l'epoca storica e le esigenze dei programmatori.

Periodo: Sistemi Mainframe

- **Applicazioni:** Orientate al calcolo scientifico
- **Interfacce:** Predominantemente testuali
- **Amministrazione:** Scripting per automatizzare compiti

Risultati di Questo Contesto:

1. **C:** Concepito per prestazioni elevate, ottimo per il calcolo scientifico, ma meno portabile.
2. **Assembly:** Estremamente veloce, ma scarsa portabilità; utilizzato per l'interfacciamento diretto con l'hardware.
3. **Shell:** Più lento, ma altamente portabile; ideale per attività di manutenzione e scripting.

Cenni storici

Durante questa epoca, i programmatori iniziano a riconoscere due importanti tendenze nell'architettura dei calcolatori:

1. **L'Avanzamento dell'Hardware:** L'hardware delle macchine si evolve in modo significativo, seguendo un ritmo simile alla legge di Moore del 1965. Questo progresso inarrestabile fornisce una potenza di calcolo crescente.
2. **Complessità in Aumento:** La crescente complessità delle architetture rende sempre più difficile scrivere codice a basso livello. La gestione dei dettagli intricati richiede una quantità significativa di tempo e sforzo.

In risposta a queste sfide, emerge il concetto di "linguaggio ad alto livello general purpose," con esempi significativi tra cui:

- **Python (Guido Van Rossum, 1991):** Famoso per la sua semplicità e leggibilità del codice.
- **Ruby (Yukihiro Matsumoto, 1993):** Con una forte enfasi sulla programmazione orientata agli oggetti.
- **Java (James Gosling, 1995):** Notorio per la portabilità del codice attraverso piattaforme diverse.

Cenni storici

Nel 1991, il World Wide Web venne inventato, segnando l'inizio di una rivoluzione nell'informatica.

L'Emergere di Nuovi Linguaggi per Applicazioni Web:

- Al fine di sviluppare applicazioni Web-based in modo più semplice rispetto al linguaggio C, si assiste all'emergere di nuovi linguaggi.
- **PHP (PHP Hypertext Preprocessor):** Nasce nel 1994 con l'obiettivo di creare pagine Web dinamiche.
- **JavaScript (Mocha):** Viene introdotto nel 1995 per essere utilizzato in Netscape, uno dei primi browser Web.

L'Adattamento dei Linguaggi Esistenti:

- Linguaggi come **C** e **Perl** vengono adattati per non perdere terreno e vengono comunemente utilizzati per la creazione di programmi CGI (Common Gateway Interface) per la gestione dinamica delle pagine Web.

Cenni storici

- PYPL <https://pypl.github.io>
- Popolarità cambia nel corso del tempo
- Influenzato da:
 - a. Mode
 - b. Esigenze (es. Web)
 - c. Politiche

Attualmente Python è il più popolare

Worldwide, Sept 2023 :

Rank	Change	Language	Share	1-year trend
1		Python	27.99 %	+0.1 %
2		Java	15.9 %	-1.1 %
3		JavaScript	9.36 %	-0.1 %
4		C#	6.67 %	-0.4 %
5		C/C++	6.54 %	+0.3 %
6		PHP	4.91 %	-0.4 %
7		R	4.4 %	+0.2 %
8		TypeScript	3.04 %	+0.2 %
9	↑↑	Swift	2.64 %	+0.6 %
10		Objective-C	2.15 %	+0.1 %
11	↑↑	Rust	2.12 %	+0.5 %
12	↓↓↓	Go	2.0 %	-0.1 %
13	↓	Kotlin	1.78 %	-0.0 %
14		Matlab	1.58 %	+0.1 %
15		Ruby	1.05 %	-0.1 %
16	↑↑↑	Ada	1.02 %	+0.3 %
17	↑	Dart	0.95 %	+0.1 %

Linguaggi Statici

Un linguaggio di programmazione statico, o "linguaggio di programmazione con tipizzazione statica", è un tipo di linguaggio di programmazione in cui i tipi delle variabili e delle espressioni sono verificati a tempo di compilazione anziché a tempo di esecuzione.

Tipicamente:

- Linguaggi ad alto livello
- Linguaggi compilati

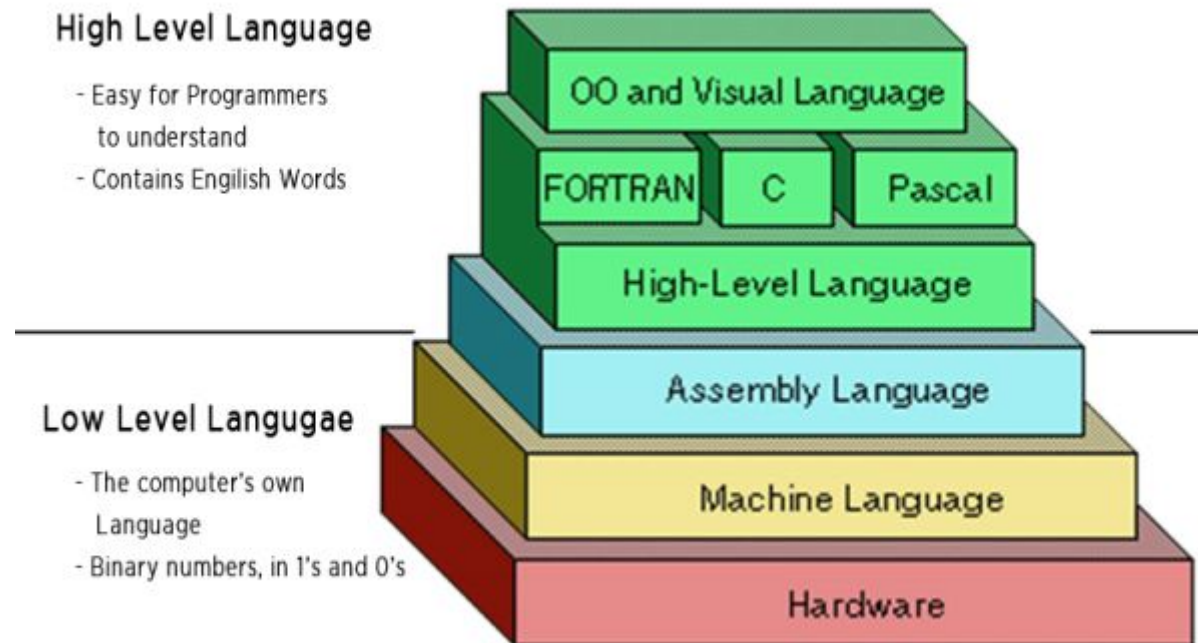
Linguaggi Dinamici

Un linguaggio di programmazione dinamico, o "linguaggio di programmazione con tipizzazione dinamica", è un tipo di linguaggio di programmazione in cui i tipi delle variabili e delle espressioni vengono verificati a tempo di esecuzione anziché a tempo di compilazione.

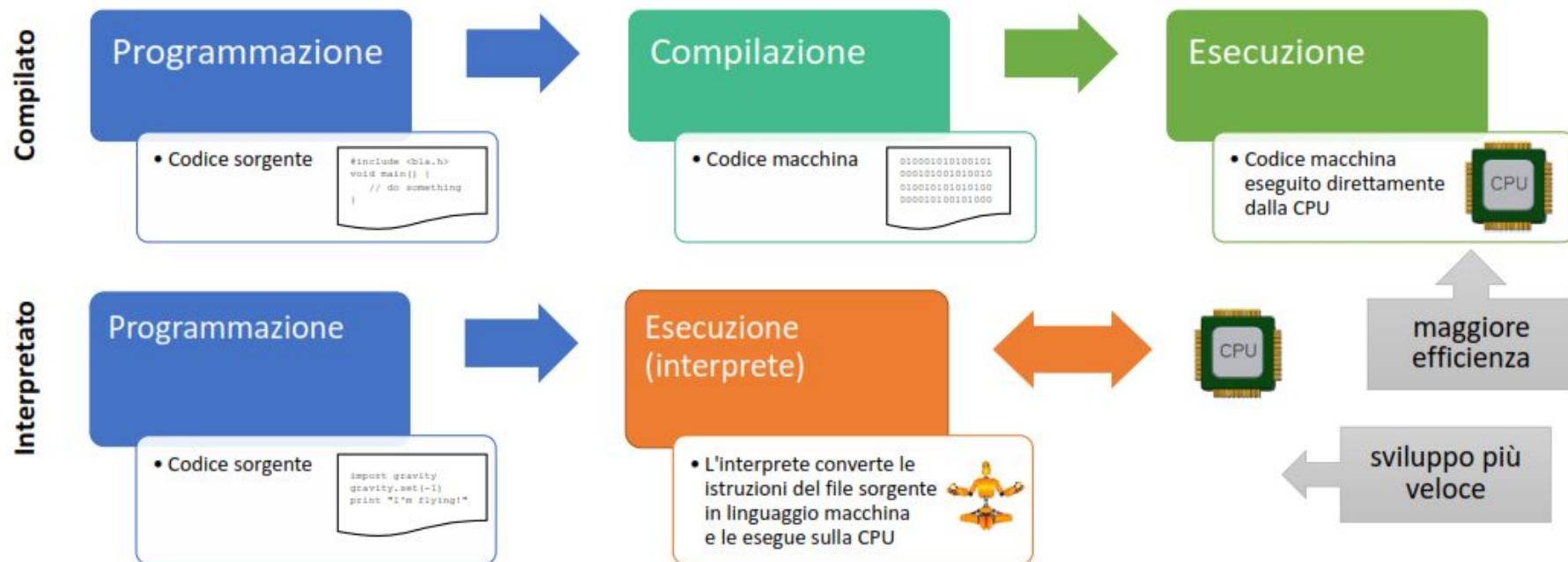
Tipicamente:

- Linguaggi ad alto livello
- Linguaggi interpretati

Linguaggi Alto livello / Basso



Linguaggi Interpretati / compilati



Pro Linguaggi Dinamici

- Tipizzazione dinamica: In un linguaggio dinamico, non è necessario dichiarare esplicitamente il tipo di una variabile durante la sua creazione. Il tipo di una variabile può cambiare durante l'esecuzione del programma.
- Controllo dei tipi a runtime: I controlli sui tipi e le conversioni vengono effettuati durante l'esecuzione del programma.
- Flessibilità: I linguaggi dinamici tendono a essere più flessibili e meno rigidi rispetto ai linguaggi con tipizzazione statica.
- Maggiore facilità di debug: Poiché i tipi vengono verificati a runtime, è possibile esaminare più facilmente il comportamento del programma durante il debug.

Pro Linguaggi Dinamici

- Sviluppo più rapido
- Tipicamente sintassi più semplice

Contro Linguaggi Dinamici

- **Prestazioni inferiori:** I linguaggi dinamici tendono ad essere più lenti dei linguaggi con tipizzazione statica. Questo perché la verifica dei tipi e l'allocazione della memoria possono comportare un overhead significativo durante l'esecuzione del programma. Le ottimizzazioni possono mitigare questo problema, ma in generale i linguaggi dinamici sono meno efficienti in termini di velocità rispetto ai linguaggi statici.
- **Errori a runtime:** Poiché i controlli dei tipi avvengono a runtime, gli errori di tipo possono emergere solo durante l'esecuzione del programma. Questo rende il debug più complesso e può comportare la scoperta di errori solo quando il programma è in esecuzione, il che può essere problematico in applicazioni critiche.
- **Difficoltà nella manutenzione del codice:** A causa della tipizzazione dinamica, la manutenzione del codice può essere più complessa. È possibile che i cambiamenti al codice siano difficili da tracciare e da comprendere, soprattutto in progetti di grandi dimensioni. La mancanza di informazioni sui tipi può rendere il codice meno auto-documentante.
- **Maggiore probabilità di bug:** A causa della mancanza di verifica statica dei tipi, è più facile commettere errori legati ai tipi in linguaggi dinamici. Questi errori possono non emergere fino a quando il programma viene eseguito, portando a comportamenti imprevisti.

Caratteristiche Linguaggi Dinamici

- **Può avere una fase di compilazione, in cui il codice sorgente viene tradotto in un formato intermedio indipendente dall'architettura (es. bytecode)**
- **Il formato intermedio è interpretato**
 - linguaggio portabile
- **L'interprete si serve di funzioni interne per gestire memoria ed errori in modo automatico a run-time**
 - Assenza di dettagli ostici per il programmatore
- **Ha una tipizzazione dinamica dei dati**
 - I tipi di dato possono mutare a run-time

Caratteristiche Linguaggi Dinamici

- **Ha la caratteristica di sapersi “analizzare” e “modificare” durante l'esecuzione (Metaprogramming)**
 - Eseguire funzioni diverse a seconda delle condizioni operative a run-time
 - Cambiare il codice stesso del programma
 - Creare strutture dati variabili nel tempo
- **Presenza massiccia di librerie esterne facilmente utilizzabili per diversi compiti**
 - Servizi di calcolo scientifico, interfacce grafiche complesse, supporto per il Web

Python perchè?

Linguaggio (dinamico) più utilizzato

Worldwide, Sept 2023 :

Rank	Change	Language	Share	1-year trend
1		Python	27.99 %	+0.1 %
2		Java	15.9 %	-1.1 %
3		JavaScript	9.36 %	-0.1 %
4		C#	6.67 %	-0.4 %
5		C/C++	6.54 %	+0.3 %
6		PHP	4.91 %	-0.4 %
7		R	4.4 %	+0.2 %
8		TypeScript	3.04 %	+0.2 %
9	↑↑	Swift	2.64 %	+0.6 %
10		Objective-C	2.15 %	+0.1 %
11	↑↑	Rust	2.12 %	+0.5 %
12	↓↓↓	Go	2.0 %	-0.1 %
13	↓	Kotlin	1.78 %	-0.0 %
14		Matlab	1.58 %	+0.1 %
15		Ruby	1.05 %	-0.1 %
16	↑↑↑↑	Ada	1.02 %	+0.3 %
17	↑	Dart	0.95 %	+0.1 %

Python perchè?

Programming languages used in most popular websites*

Websites ⇅	Popularity (unique visitors per month) ^[1] ⇅	Front-end (Client- side) ⇅	Back-end (Server-side) ⇅	Database ⇅	Notes
Google ^[2]	2,800,000,000	JavaScript, TypeScript	C, C++, Go, ^[3] Java, Python, Node	Bigtable, ^[4] MariaDB ^[5]	The most used search engine in the world.
Facebook	1,120,000,000	JavaScript, Typescript, Flow	Hack/HHVM, Python, C++, Java, Erlang, D, ^[6] Haskell ^[7]	MariaDB, MySQL, ^[8] HBase, Cassandra ^[9]	The most visited social networking site.
YouTube	1,100,000,000	JavaScript, TypeScript	Python, C, C++, Java, ^[10] Go ^[11]	Vitess, BigTable, MariaDB ^[5] ^[12]	The most popular video sharing site.

Python perchè?

Popolarità in ascesa

E' il linguaggio dinamico di riferimento per la piattaforma dei servizi offerti da Google

Curva di apprendimento ripida

Flessibilità

Semplicità

Python perchè?

C++

```
1  
2#include<iostream>  
3using namespace std;  
4  
5int main (int argc, int *argv[])  
6{  
7    cout << "Hello World" << endl;  
8    return 0;  
9}
```

● Complex syntax

Python

```
1  
2 print "Hello World"
```

● Minimal Syntax