

GIUSEPPE CHELLINI

Introduzione

Un pò di storia...

Linguaggi per la  
programmazione

Linguaggi per la  
matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

Software usato per la  
presentazione

# Software libero per il calcolo matematico

## Strumenti FOSS per l'Ingegneria

GIUSEPPE CHELLINI

GULP - Gruppo Utenti Linux Pisa

26 Ottobre 2019



[Introduzione](#)[Un pò di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Un pò di storia...

## Ingegneria e Matematica: inizio '900...

L'Ingegneria è fatta di buone regole dell'arte, di esperienza e di pratica... la matematica è solo una perdita di tempo

## Ingegneria e Matematica: anni 50-70...

L'Ingegneria è fortemente assistita dalla matematica... peccato che sia così faticoso fare i calcoli. L'ingegnere fa del suo meglio per effettuarli!

## Ingegneria e Matematica: anni 80 in poi...

L'Ingegneria è fortemente assistita dalla matematica! Grazie ai computer è diventato semplice fare i conti...

[Introduzione](#)[Un po' di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Il calcolo alle origini...

La possibilità di effettuare calcoli "non a mano" (es. con il regolo calcolatore) ma mediante **calcolatrici** prima e **computer** tipo Mainframe poi ha permesso di affrontare progettazioni prima non affrontabili.



Di Robocomm - Opera propria

CC BY-SA 3.0

Ad esempio, a Pisa la presenza del **CNUCE** (*Centro Nazionale Universitario di Calcolo Elettronico*) ha permesso a molti ricercatori ed ingegneri di utilizzare gli algoritmi numerici noti all'epoca per lo studio delle sollecitazioni nelle strutture.



[Introduzione](#)[Un po' di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Il calcolo finalmente accessibile...



Sinclair ZX Spectrum 48K



Commodore 64

[Vedere 8-Bit Generation ...](#)



Computer fisso con Win98

Questi strumenti hanno permesso l'uso della programmazione anche in ambiente domestico.

[Introduzione](#)[Un po' di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Il calcolo alla portata di tutti...



Laptop



Computer monoscheda low-cost



Mobile

Molti di questi strumenti sono presenti nelle nostre case ed hanno una potenza di calcolo molto superiore (o.d.g.) dei vecchi sistemi.

[Introduzione](#)[Un pò di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Come pensa un computer?

Il computer "pensa" in **Linguaggio Macchina**...

- Il linguaggio macchina è basato su un alfabeto binario (0, 1).
- Il **processore** è quel **componente hardware** che è in grado di **eseguire i programmi** scritti in linguaggio macchina.
- In linguaggio macchina sono definite l'**insieme di istruzioni fondamentali** che un processore è in grado di eseguire (*instruction set*).
- Nel linguaggio macchina i simboli (0, 1) sono organizzati in "parole" che a loro volta costituiscono "frasi". Le frasi del linguaggio macchina sono dette istruzioni: ognuna di esse ordina al processore di eseguire un'azione elementare afferente allo stato interno del computer.

[ op   rs   rt   rd  shamt  funct]						
0	1	2	6	0	32	forma decimale
000000	00001	00010	00110	00000	100000	forma binaria

[Introduzione](#)[Un pò di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Come si parla con un computer?

Il set delle istruzioni elementari non è standard ed è **limitato**. Programmare in linguaggio macchina è per noi "quasi" impossibile... sono state individuate due possibili soluzioni al problema:

**Compilatore:** un programma che **traduce** una serie di istruzioni scritte in un determinato linguaggio di programmazione (**codice sorgente**) in istruzioni di un altro linguaggio (**codice oggetto**);

**Interprete:** un programma **in grado di eseguire altri programmi** a partire direttamente dal relativo codice sorgente. Ha lo scopo di eseguire un programma in un linguaggio di alto livello, senza la previa compilazione dello stesso (**codice oggetto**).

[Introduzione](#)[Un po' di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Compilatori o Interpreti

- Le soluzioni basate sull'utilizzo di compilatori sono generalmente più "veloci": la compilazione è fatta una volta sul codice sorgente ed il codice oggetto ottenuto è sempre eseguito direttamente;
- Le soluzioni basate sull'utilizzo di interpreti consentono una maggiore flessibilità ma producono programmi generalmente più "lenti"; (*Ciclo FDE: Fetch Decode Execute*)
- Sono state realizzate delle soluzioni "ibride" che, ad esempio,
  - compilano il codice sorgente in un linguaggio pronto per essere interpretato (es. Java Virtual Machine e .NET)
  - effettuano la compiazione durante l'esecuzione del programma piuttosto che prima (**JIT JustInTime**)

## Introduzione

Un po' di storia...

## Linguaggi per la programmazione

## Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

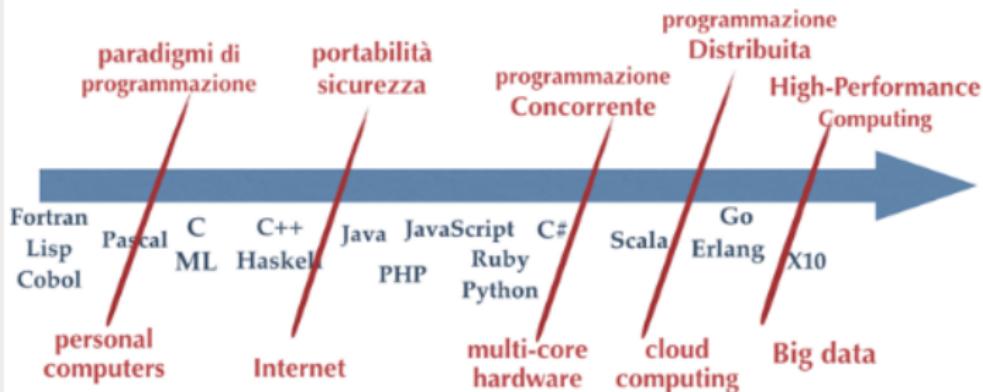
Big Data

## Software usato per la presentazione



# Linguaggi di programmazione

## *Salti Evolutivi*



## *Catalizzatori*

[ Tratta da Mondo Digitale - Feb 2016]

# C/C++

## Introduzione

Un pò di storia...

## Linguaggi per la programmazione

### Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

## Software usato per la presentazione



## Caratteristiche

- Compilato (es. GCC - GNU Compiler Collection)

## Pros

- Elevata velocità computazionale
- Programmi si integrano bene con OS (es. drivers)
- Esistono molte librerie consolidate
- "Community" di utilizzatori vasta
- Esistono numerose piattaforme FOSS per sviluppo e debug

## Cons

- Basso livello di astrazione
- Ridotta portabilità.

# Python

## Introduzione

Un pò di storia...

## Linguaggi per la programmazione

## Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

## Software usato per la presentazione

## Caratteristiche

- Interpretato

## Pros

- Elevata portabilità
- Esistono numerose librerie per molte attività
- "Community" di utilizzatori vasta
- Linguaggio di alto livello con elevato potere espressivo
- Esistono numerosi ambienti di sviluppo avanzati FOSS

## Cons

- Ridotta velocità computazionale



# Java

## Introduzione

Un pò di storia...

## Linguaggi per la programmazione

### Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

### Software usato per la presentazione

## Caratteristiche

- Interpretato da JVM dopo compilazione del sorgente

## Pros

- Elevata portabilità ma con JVM non libera ...
- Esistono numerose librerie per molte attività
- "Community" di utilizzatori vasta
- Linguaggio di alto livello con elevato potere espressivo
- Esistono numerosi ambienti di sviluppo avanzati FOSS

## Cons

- Sono stati sviluppati strumenti di sviluppo liberi (OpenJDK) ma Java è di Oracle



# Julia

## Introduzione

Un pò di storia...

## Linguaggi per la programmazione

## Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

## Software usato per la presentazione



## Caratteristiche

- JIT - JustInTime Compilation

## Pros

- Elevata velocità computazionale
- Linguaggio di alto livello con elevato potere espressivo

## Cons

- "Community" di utilizzatori ridotta ma in crescita
- Esistono librerie per molte attività ma in numero ridotto rispetto Python

[Introduzione](#)[Un po' di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Linguaggi per la matematica

Utilizzando i linguaggi precedentemente descritti sono stati sviluppati applicativi (tipicamente interpreti) per usi matematici. Alcune delle tipologie di applicativi disponibili:

- applicazioni per la manipolazione di **espressioni simboliche e numeriche** (es. differenziazione, integrazione, sistemi di equazioni, matrici, ...);
- applicazioni per la **simulazione numerica** di sistemi fisici complessi (es. statici o dinamici, lineari o non lineari, ...)
- applicazioni per il **trattamento ed analisi di dati** (es. segnali, immagini, ...)
- applicazioni per il trattamento ed analisi di **grandi volumi di dati (Big Data)** (risultati di monitoraggi, ricerche statistiche, ...)

**Introduzione**

Un po' di storia...

**Linguaggi per la programmazione****Linguaggi per la matematica**

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

**Software usato per la presentazione**

# Computer Algebra System (CAS)

## Obiettivo

- manipolazione di espressioni simboliche e numeriche

## Applicativi

- wxMaxima è un Computer Algebra System (CAS) in grado di eseguire calcoli numerici, simbolici, grafici e altre operazioni correlate.

The screenshot shows two instances of the wxMaxima application. The left window displays a plot of the function  $y = x^3 + x + 1$  from -4 to 4, with a red horizontal line at  $y = 0$ . The right window shows a series of symbolic calculations:

```

(%i1) expr: (x^3+x+1)*exp(-x^2/2);
(%o2) (x^3+x+1) * e^-x^2/2
(%i3) wxplotdf(expr, 4), [x, -4, 4], [legend, "", ""],
[gnuplot_preamble, "set grid"]\$

(%i4) find_root(expr=0, x, -1, 0);
(%o4) - .34734801904092

(%i5) find_root(expr=0, x, 1, 2);
(%o5) 1.576139766829181

```

The right window also contains other calculations and plots:

```

(%i6) 4*x^1/3;
(%o6) 16/3
(%i7) (x^3+x)^2;
f(1.34);
(%o7) f(x):=x^2
(%o8) 1.795
(%i9) integrate(1/(1+x^3),x);
(%o9) - log(x^2-x+1) + atan((x-1)/sqrt(3)) + log(x+1)
(%i10) A:matrix([1,2],[3,4]);
(%o10) [[1, 2], [3, 4]]
(%i12) %e10^2;
(%o12) [2, 4]
(%i14) plot2d(sin(x),[x,-5,5]);
(%o14) 

```

**Introduzione**

Un pò di storia...

**Linguaggi per la programmazione****Linguaggi per la matematica**

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

**Software usato per la presentazione**

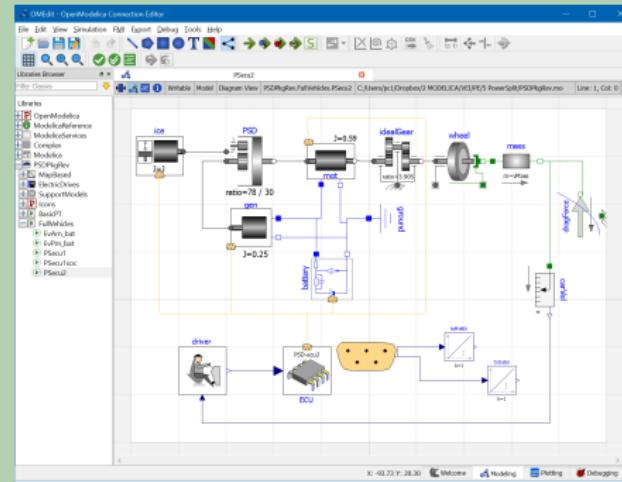
# Simulazione

## Obiettivo

- **simulazione numerica di sistemi fisici complessi**

## Applicativi

- **OpenModelica: FOSS [Modelica Language]**



Introduzione

Un po' di storia...

Linguaggi per la programmazione

Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

Software usato per la presentazione



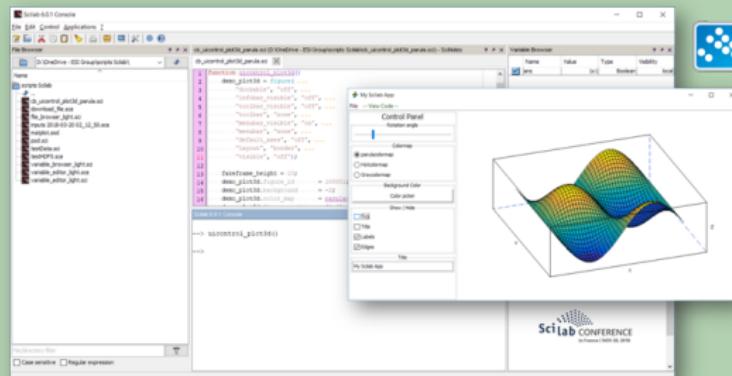
# Simulazione / Analisi Dati

## Obiettivo

- **simulazione numerica** di sistemi fisici complessi
- **trattamento/analisi di dati** (es. segnali, immagini, ...)

## Applicativi

- **SciLab [MatLab replacement]**



Introduzione

Un pò di storia..

## Linguaggi per la programmazione

## Linguaggi per la matematica

Computer Algebra System (CAS)

Simulazione

Big Data

## Software usato per la presentazione



## Simulazione / Analisi Dati

# Obiettivo

- **simulazione numerica** di sistemi fisici complessi
  - **trattamento/analisi di dati** (es. segnali, immagini, ...)

## Applicativi

- **Octave** [MatLab replacement]

[Introduzione](#)[Un pò di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Big Data

## Obiettivo

- trattamento/analisi di **grandi volumi di dati (Big Data)**

## Applicativi

- **Python** insieme a librerie specifiche



NumPy



SciPy



matplotlib

**IP[y]: IPython**  
Interactive Computing

pandas

$$y_t = \beta' x_t + \mu_t + \epsilon_t$$



[Introduzione](#)[Un pò di storia...](#)[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)[Computer Algebra System \(CAS\)](#)[Simulazione](#)[Simulazione / Analisi Dati](#)[Big Data](#)[Software usato per la presentazione](#)

# Big Data

## Obiettivo

- trattamento/analisi di grandi volumi di dati (Big Data)

## Applicativi

- Julia insieme a librerie specifiche



[Introduzione](#)

Un pò di storia...

[Linguaggi per la programmazione](#)[Linguaggi per la matematica](#)

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

[Software usato per la presentazione](#)

# Software per la presentazione

## IMPORTANTE!!!

La presentazione è stata realizzata interamente  
mediante SOFTWARE LIBERO.

- Latex con pacchetto Beamer  
(<https://www.ctan.org/tex-archive/macros/latex/contrib/beamer>)
- editor per LaTeX: TexStudio (<http://www.texstudio.org/>)
- Beamer Template: Laughlin by Mohamed El Morabity  
([https://fedoraproject.org/wiki/Templates\\_for\\_Presentations](https://fedoraproject.org/wiki/Templates_for_Presentations))
- Info descrittive: Wikipedia  
([https://it.wikipedia.org/wiki/Pagina\\_principale](https://it.wikipedia.org/wiki/Pagina_principale))

# Saluti

Introduzione

Un pò di storia...

Linguaggi per la  
programmazione

Linguaggi per la  
matematica

Computer Algebra System (CAS)

Simulazione

Simulazione / Analisi Dati

Big Data

Software usato per la  
presentazione

**Domande?**

**Buon Linux Day!**

