

Introduzione a MATLAB

- Informazioni sul laboratorio
- MATLAB
 - Operazioni di base
 - Variabili ed assegnamento
 - Comandi utili
 - Utilizzo dell'Help
 - Formato dei numeri
 - Numeri complessi

1

Il Laboratorio

- Openlab - Ultimo Piano
- 64 PC – Windows 2000
- Per entrare:
 - Username: **utente**
 - Password: **Infonew1**
- Possiamo scrivere solo sulla directory c:\Temp , che viene cancellata ad ogni riavvio.
- Quindi è utile avere una chiavetta USB, o un floppy per salvare il proprio lavoro. (O spedirlo via mail)

2

Oltre alle lezioni...

- Potete accedere al laboratorio ogni volta che trovate posto
- Compatibilmente con lezioni ed esami
- Matlab non è free (non potete averlo a casa)
- Però ci sono dei cloni gratuiti:
 - Octave (www.octave.org)
 - Scilab (www.scilab.org)

3

MATLAB – Cos'è ?

- Il nome deriva da **matrix** **laboratory**
- Strumento potente per:
 - Calcolo e visualizzazione di problemi scientifici e di ingegneria
 - Scambiare metodi e idee
 - Per programmare:
 - Editor, debugger, e help inclusi
 - Molte funzioni predefinite (*toolboxes*)
 - Programmi interpretati o compilati

4

Storia di Matlab

- Pacchetti LINPACK, EISPACK per fortran scritti da Cleve Moler negli anni 70.
- Sistema interattivo per la didattica
- Riscritto in C negli anni '80, ora in java
- Passa-parola inizialmente, ora c'è la Mathworks, inc

5

Pro e contro di Matlab

PRO

- Facile da imparare
- Ottimizzato per i calcolo vettoriale
- Doppio uso:
 - Command window
 - Programmi
- Facile da correggere perché interpretato

CONTRO

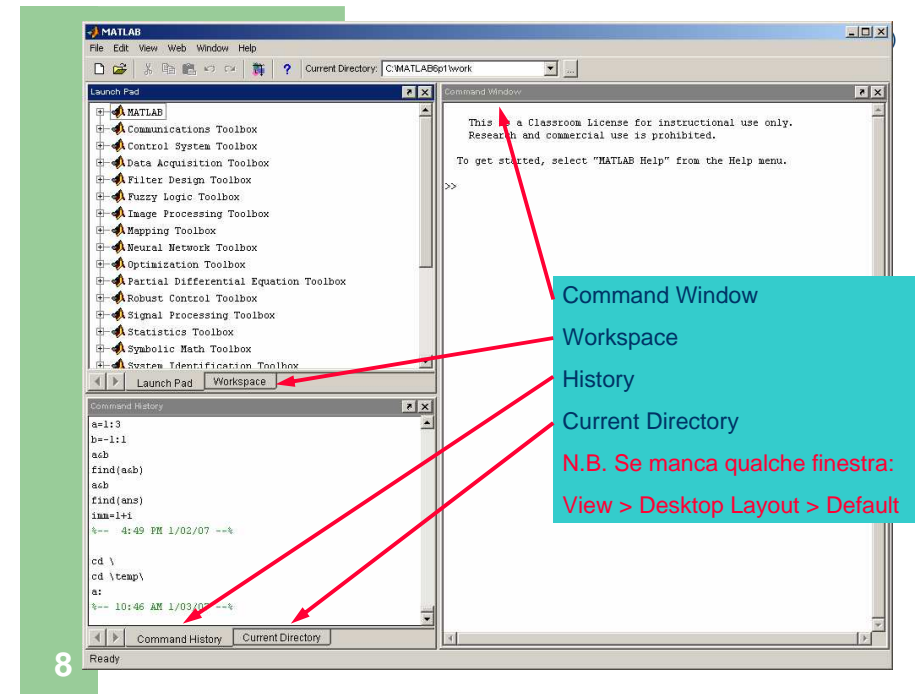
- Lento rispetto ai linguaggi compilati
- NON general purpose
- Trattamento dei testi limitato

6

L'ambiente MATLAB

- Dati raccolti in vettori
 - Organizzati in righe e colonne con indici
 - Utilizzo di nomi di variabili
- Ambiente multi-pannello:
 - Command window
 - Command history
 - Workspace browser
 - Current directory
 - Altre finestre: Figure, File Editor, Help,

7



8

Command Window

- Command prompt **>>**
- A disposizione le operazioni di base:
 addizione **+** sottrazione **-** divisione **/**
 moltiplicazione ***** potenza **^** div.inv. ****
 il tasto **"enter"** esegue o "lancia" l'operazione
- Precedenza degli operatori: PE(MD)(AS)
 $5 - 4 + 3^4 / (3 - 1) = ?$
- In caso di pari precedenza, esegue da sinistra a destra

9

```
Command Window
>> 2 + 4

ans =

     6

>> 3 * 7

ans =

    21

>> 4 / 3

ans =

    1.3333
```

```
Command Window
>> 2^4

ans =

    16

>> 7 / 0
Warning: Divide by zero.

ans =

    Inf

>> 3 + 4 * 2 - 5

ans =

     6
```

10

- Precisione finita!!
- Di default, 5 cifre significative (con notazione esponenziale se necessario)
- Possibili risultati
NaN, Inf

```
Command Window
>> 1 - .2 - .2 - .2 - .2 - .2

ans =

    5.5511e-017

>> 0 / 0
Warning: Divide by zero.

ans =

    NaN

>> 7 / 0
Warning: Divide by zero.

ans =

    Inf
```

11

Variabili consentite

- L'uguale è un operatore di **assegnamento**
c = 7.5 bob3 = 3.7789
- Alcune regole per i nomi:
 - Iniziare con una lettera e niente spazi
 - Unici!!!
 - Alcuni nomi sono riservati a priori:
pi inf eps ans (j i)

12

Combinare Operazioni e =

- Generalmente il calcolo richiede tre informazioni:

- Che operatore?
- Che input?
- Dove mettere l'output?
 - MATLAB → mettiamo il risultato in una variabile (ans)
 - Si utilizza l'assegnamento, =
 - A sinistra dell'operatore = deve esserci una singola variabile.
 - A destra deve esserci un valore *calcolabile*

Questi vanno a destra del simbolo di assegnamento

13

In questi esempi gli input sono 2 e 4 e l'operazione è la somma

```
Command Window
>> 2 + 4
ans =
    6
>> bob = 2 + 4
bob =
    6
>> bob = plus(2,4)
bob =
    6
```

Non specifichiamo la variabile di output, di default il risultato vien messo nella variabile ans

Mettiamo il risultato nella variabile bob

Mettiamo sempre il risultato in bob, ma abbiamo svolto il calcolo con la *notazione funzionale*

14

Il simbolo = non è un confronto, è un assegnamento!

```
Command Window
>> bob = 6
bob =
    6
>> bob = bob + 1
bob =
    7
>> 3 + 4 = bob
??? 3 + 4 = bob
Error: The expression to the
```

Possiamo anche non avere calcoli a sinistra

Possiamo utilizzare nei calcoli la stessa variabile in cui metteremo poi il risultato!

Qua abbiamo scambiato le due parti, ERRORE!!!

15

Altri operatori utili

abs(x)	sign(x)	log(x)	exp(x)
round(x)	floor(x)	ceil(x)	fix(x)
sqrt(x)	conj(x)	log10(x)	rem(x,y)
sin(x)	asin(x)	sinh(x)	asinh(x)
cos(x)	acos(x)	cosh(x)	acosh(x)
tan(x)	atan(x)	atan2(x,y)	

- E tanti altri !!
- Per scoprirne il significato: `help <nome>`

16

Esempi

```

Command Window
>> hypot = sqrt ( 3^2 + 4^2 )
hypot =
    5
>> angle = asin(1)
angle =
    1.5708
>> root = 2 + sqrt( -5 )
root =
    2.0000 + 2.2361i

```

Radice quadrata

Le funzioni trigonometriche usano i radianti!!!

Possiamo passare facilmente ai numeri complessi

17

Varie ed eventuali...

- Il punto e virgola (;) dopo un comando non visualizza l'output
- Tre puntini (...) consentono di continuare il comando nella riga successiva
- Altri comandi di sistema:
 - `help`, `lookfor`, `doc`
 - `clear`, `clc`
 - `who`, `whos`
- **Tasti freccia** per copiare o modificare i comandi passati

18

Comandi di help

- `help funcname`: Visualizza nella finestra di comando una descrizione della funzione `funcname`.
- `lookfor topic`: Visualizza nella finestra di comando una breve descrizione di tutte le funzioni che includono la parola chiave `topic` nella descrizione.
- `doc funcname`: Apre l'help alla pagina del manuale per la funzione `funcname`, fornendo una descrizione, note aggiuntive ed esempi.

19

L'Help Browser di MATLAB :

- *Contents*: Un sommario ad albero di tutti i contenuti,
- *Index*: un indice analitico globale
- *Search*: un motore di ricerca per trovare il termine inserito, sia nel titolo che nel testo dell'help.
- *Favorites*: help addizionale via web, e raccolta di argomenti personalizzabile dall'utente.

20

Comandi utili nella sessione di lavoro

Comando	Descrizione
<code>clc</code>	Pulisce la Command window.
<code>clear</code>	Cancella tutte le variabili in memoria.
<code>clear v1 v2</code>	Cancella le variabili <code>v1</code> e <code>v2</code> dalla memoria.
<code>exist('var')</code>	Determina se esiste un file o una variabile di nome <code>'var'</code> .

21

(continua ...)

Comandi utili nella sessione di lavoro

Comando	Descrizione
<code>who</code>	Elenca le variabili attualmente in memoria.
<code>whos</code>	Elenca le variabili, le loro dimensioni e il loro tipo (come finestra Workspace)
<code>quit</code>	Esci da MATLAB.

22

Command History

- MATLAB tiene in memoria i comandi precedenti
- Con le frecce \uparrow e \downarrow , possiamo tornare tra i comandi eseguiti
- Possiamo modificare un comando (con \leftarrow , \rightarrow , *Canc* e *Backspace*)

23

Completamento automatico

- Digitando le prime lettere di una variabile o funzione e premendo *tab*:
 - Se la scelta è univoca, MATLAB completa il nome
 - Altrimenti ci elenca le possibilità
- Digitando le prime lettere di un comando, poi con le frecce \uparrow e \downarrow , possiamo tornare tra i comandi eseguiti, che iniziano con tali lettere

24

Formato dei numeri

<code>format short</code>	Quattro cifre decimali (default); 13.6745.
<code>format long</code>	16 cifre; 17.27484029463547.
<code>format short e</code>	Cinque cifre (quattro decimali) più l'esponente 6.3792e+03.
<code>format long e</code>	16 cifre (15 decimali) più l'esponente; 6.379243784781294e-04.

25

Numeri complessi

Il numero complesso $c_1 = 1 - 2i$ viene inserito in questo modo: `c1 = 1-2i`.

L'asterisco non è necessario tra i (o j) e un numero, ma è obbligatorio se segue una variabile, come:

`c2 = 5 - i*c1`.

Attenzione!!! Le espressioni

`y = 7/2*i` e `x = 7/2i`

danno due risultati diversi:

$y = (7/2)i = 3.5i$ e $x = 7/(2i) = -3.5i$.

26