Università di Salerno
Dipartimento di
Ingegneria Industriale

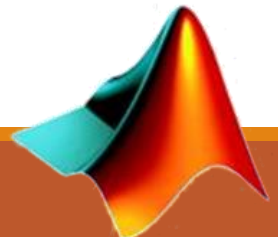
UNIVERSITÀ DEGLI STUDI DI SALERNO

Fondamenti di Informatica

Cenni di Debugging in MATLAB

Prof. Raffaele Pizzolante

A.A. 2016/17

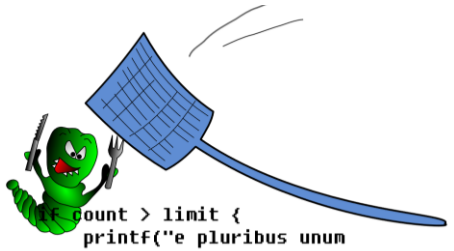


MATLAB

Debugging – 1/7

- Il **debugging** (o semplicemente debug) è l'attività che permette al programmatore di individuare la porzione di programma (codice sorgente) in cui è contenuto un errore o **bug** (bug letteralmente significa **insetto**), al fine di correggerlo

- MATLAB fornisce un *Debugger*
  - Strumento per supportare il programmatore durante la fase di debugging (ovvero per l'individuazione di errori/bug)



Debugging – 2/7

- MATLAB permette di isolare e valutare singolarmente ***sezioni (celle)*** di codice, al fine di restringere la ricerca di eventuali bug soltanto a tali sezioni
  - Questa tecnica viene detta ***modalità cella***

- Una ***sezione (o cella)*** di codice valutata dal debugger inizia con **%%**
  - **N.B.**: Da non confondere con i **commenti**, che iniziano con **%**

Debugging – 3/7

Editor – untitled3*

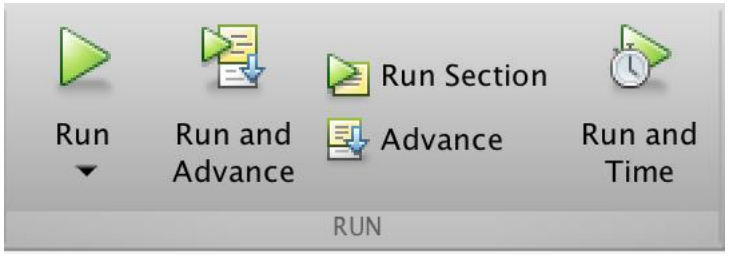
speed.m | untitled3*

```
1
2   %%
3   clear, clc;
4   x = linspace(0, 10, 300);
5   %% Funzione Quadratica
6   y1 = 2.*x.^2 + 1.*x + 3;
7   %% Funzione Cubica
8   y2 = 4.*x.^3-2.*x.^2 + 3;
```

- Questa porzione di codice è composta da **tre sezioni**
  - MATLAB separa visivamente ogni sezione mediante linee orizzontali

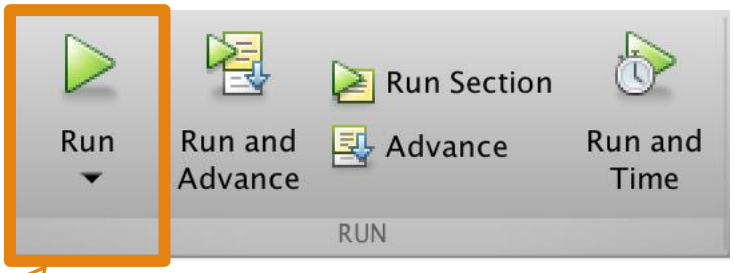
Debugging – 4/7

- È possibile eseguire la porzione di codice
  - Interamente
  - Sezione per sezione
  - Solo una determinata sezione



Debugging – 4/7

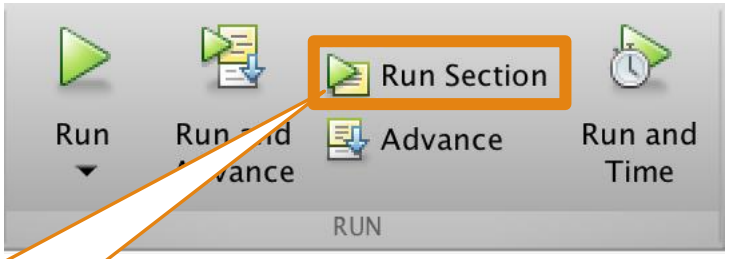
- È possibile eseguire la porzione di codice
  - Interamente
  - Sezione per sezione
  - Solo una determinata sezione



Esegue l'intera porzione di codice

Debugging – 4/7

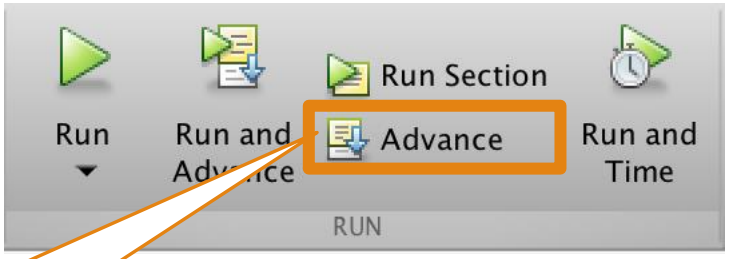
- È possibile eseguire la porzione di codice
  - Interamente
  - Sezione per sezione
  - Solo una determinata sezione



Esegue solo la sezione selezionata, o corrente

Debugging – 4/7

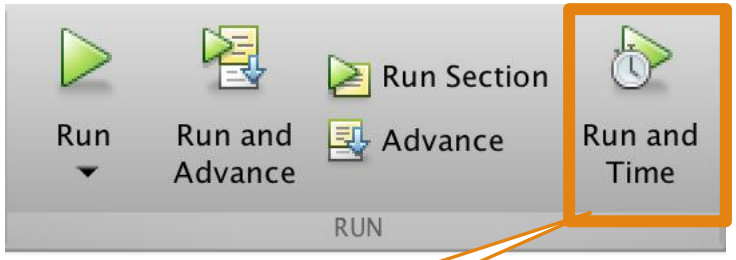
- È possibile eseguire la porzione di codice
  - Interamente
  - Sezione per sezione
  - Solo una determinata sezione



Sposta la selezione alla prossima sezione

Debugging – 4/7

- È possibile eseguire la porzione di codice
  - Interamente
  - Sezione per sezione
  - Solo una determinata sezione



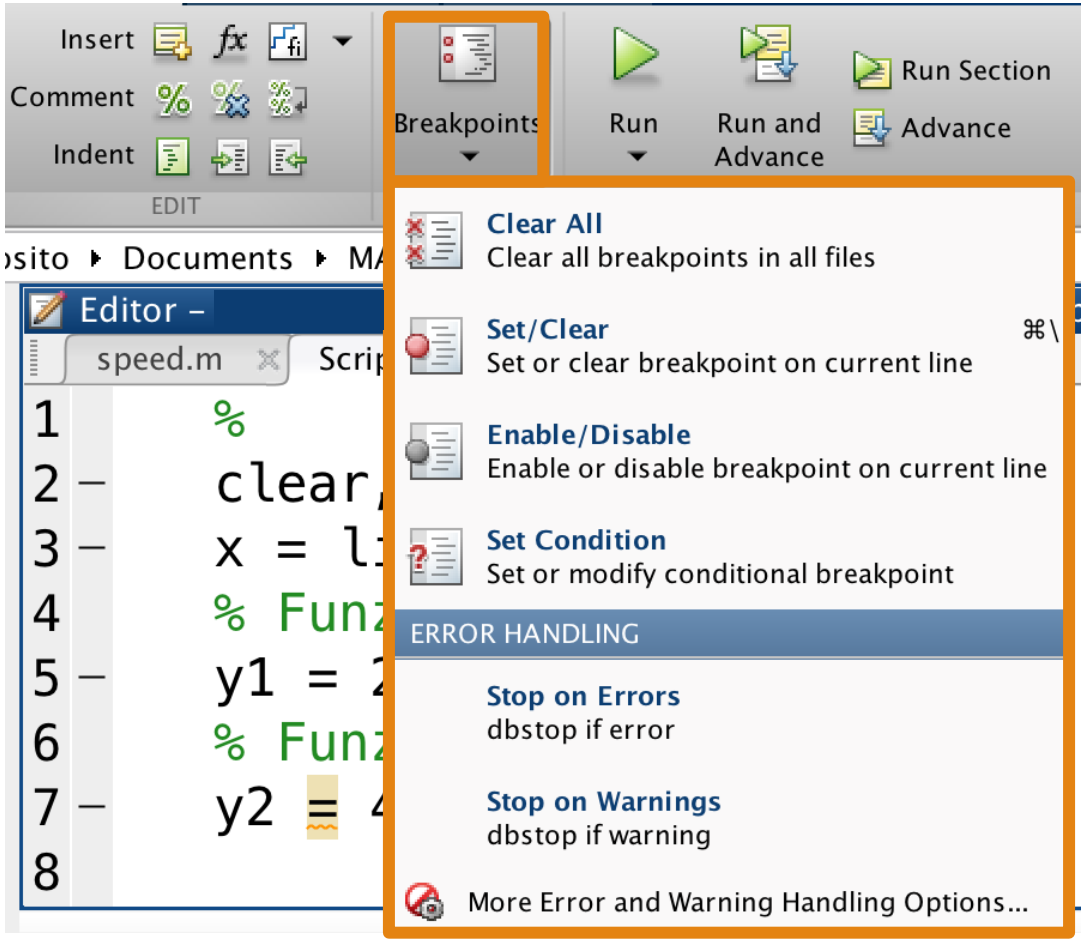
Esegue tutto il codice e misura il relativo tempo di esecuzione

Debugging – 5/7

- La ***modalità debug*** è un altro meccanismo per supportare il debugging di codice MATLAB
  - Consiste nel collocare dei *breakpoints* (letteralmente *punti di rottura*) all'interno di tale codice

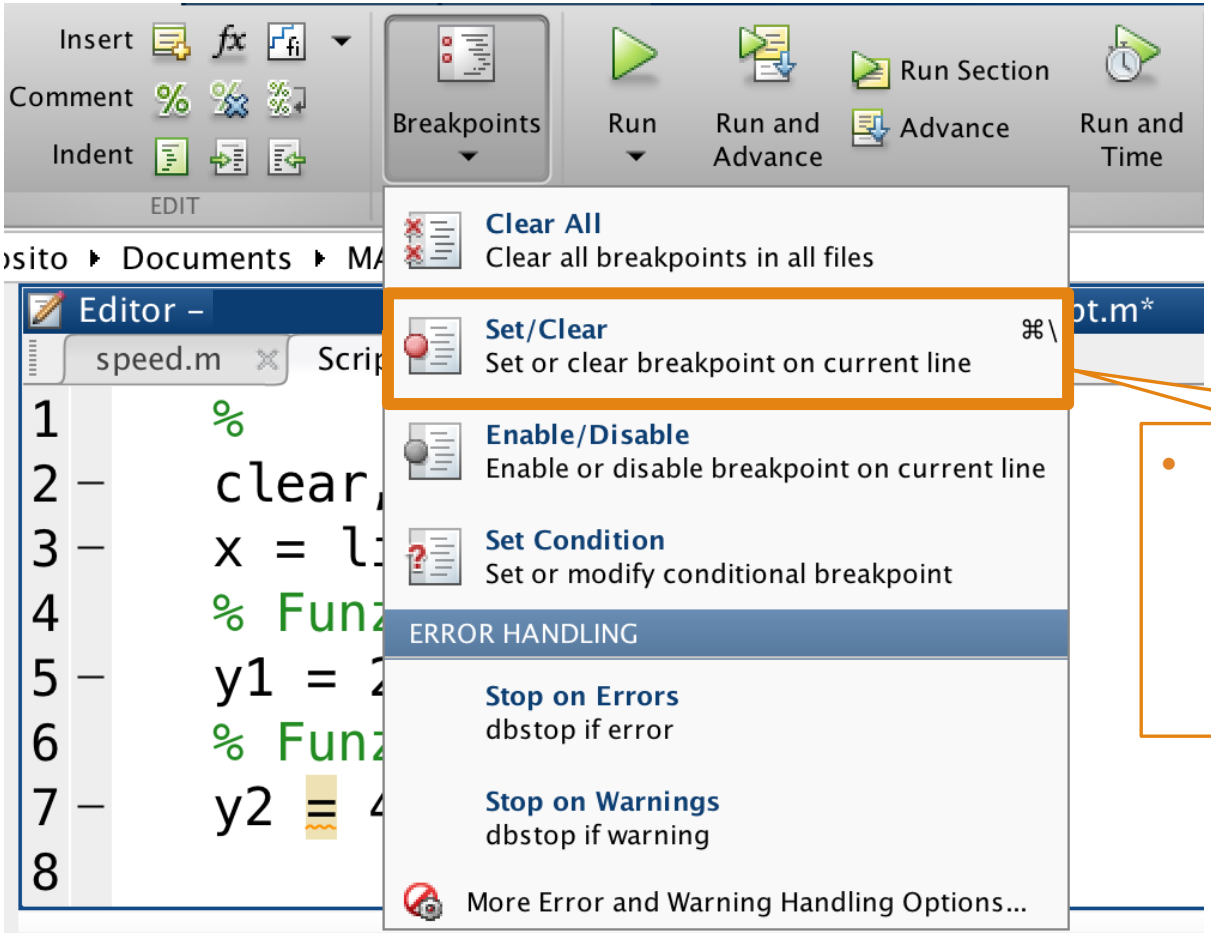
- Un *breakpoint* è un punto (riga di codice) in cui l'esecuzione del programma si interrompe temporaneamente (una sorta di pausa)
  - In tal modo è possibile esaminare i valori correnti (ed intermedi) delle variabili, semplificando l'individuazione di eventuali bug

Debugging – 6/7



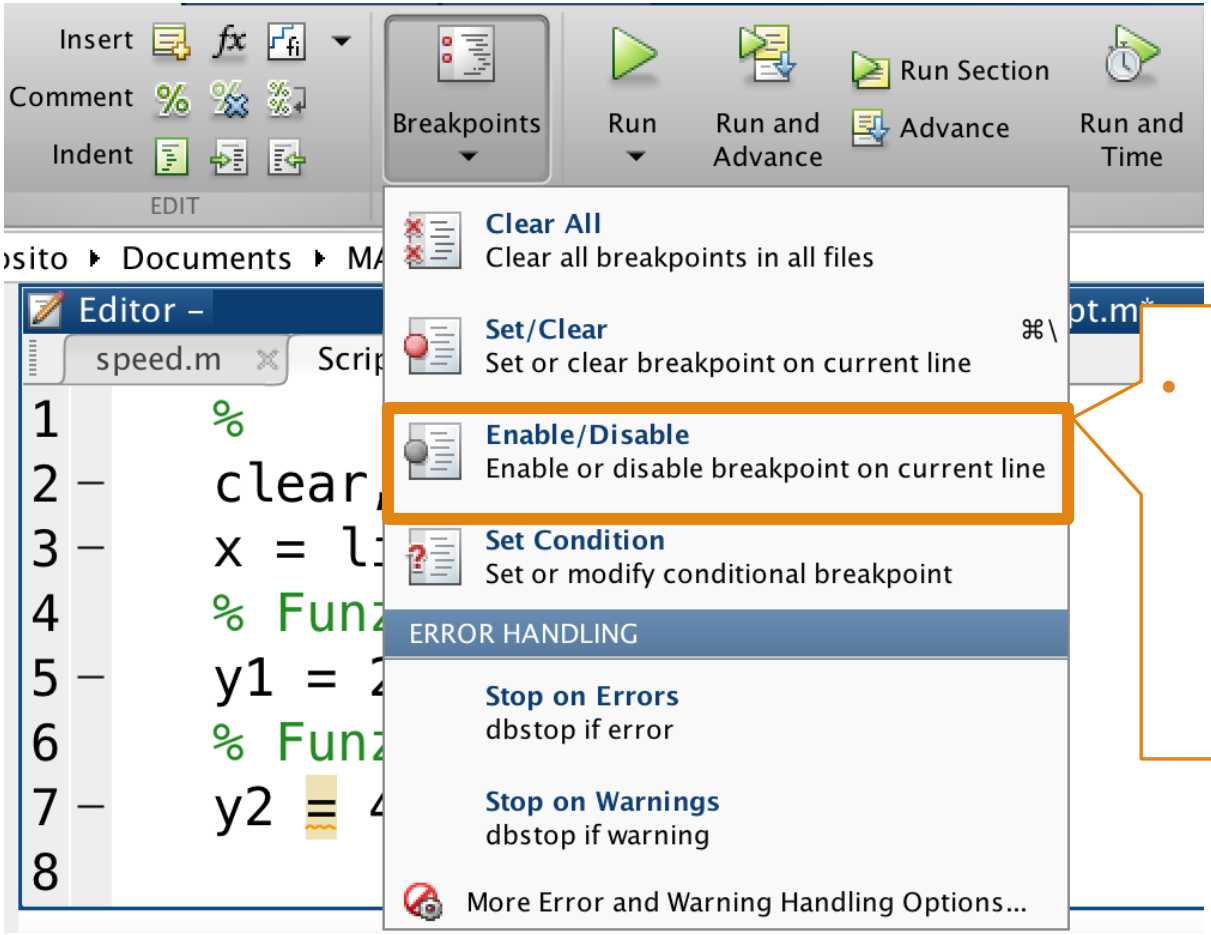
L'editor di MATLAB fornisce un apposito menù per la gestione dei *breakpoints* all'interno di un dato programma

Debugging – 6/7



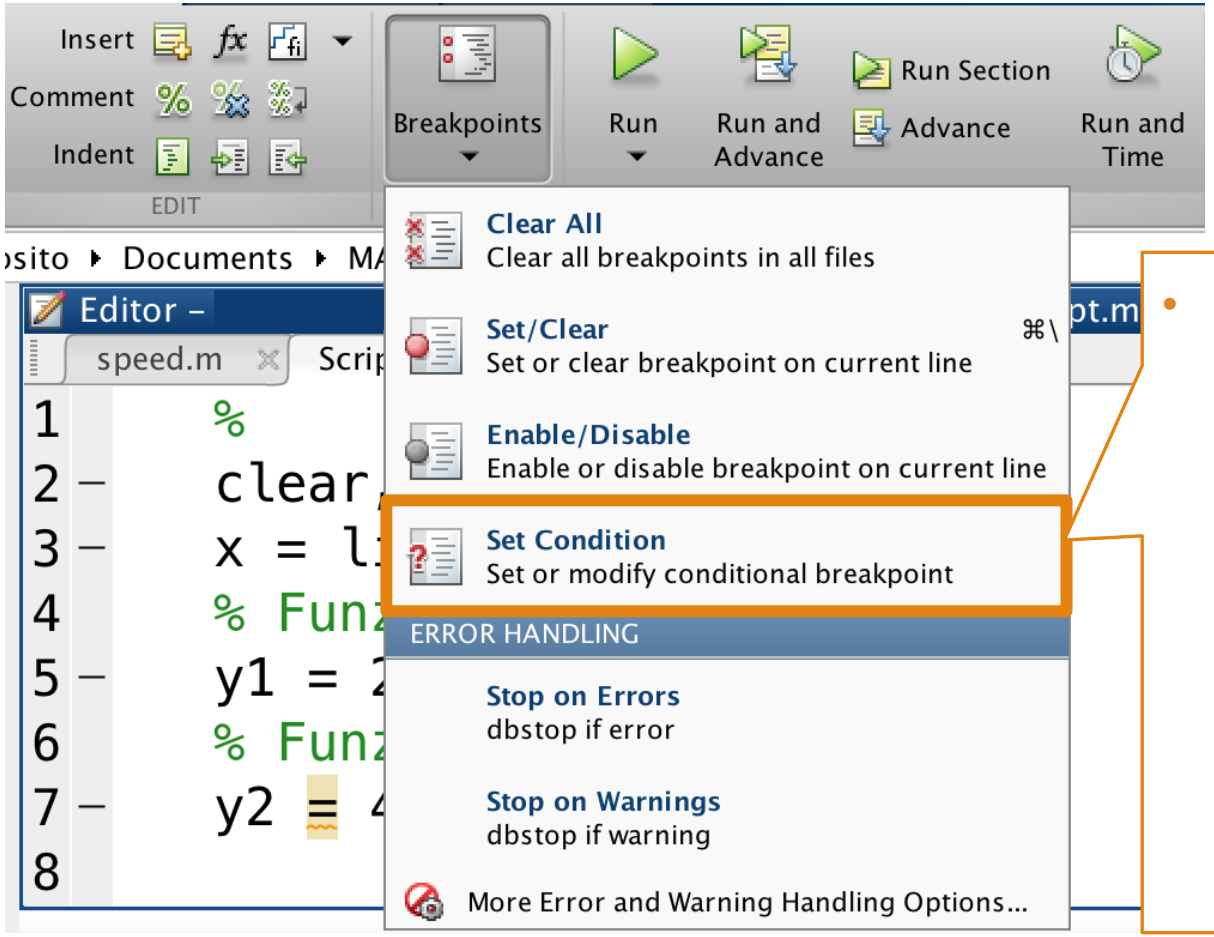
- Inserisce o rimuove un *breakpoint* alla riga corrente
  - Ciascuna riga è selezionata mediante il cursore del mouse

Debugging – 6/7



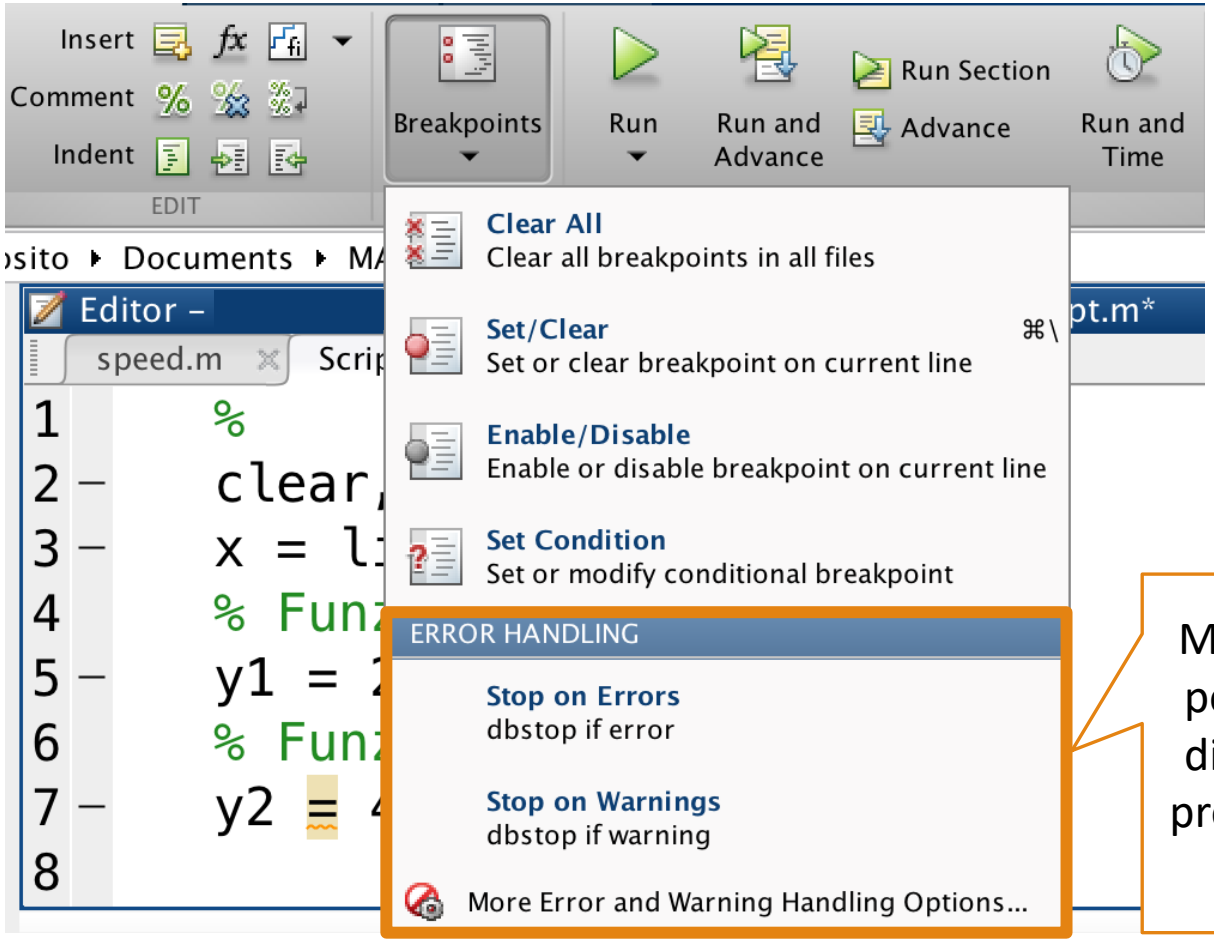
- Abilita (o disabilita) il *breakpoint* collocato alla riga corrente
  - Quando un *breakpoint* è disabilitato, esso è ignorato durante l'esecuzione del programma

Debugging – 6/7



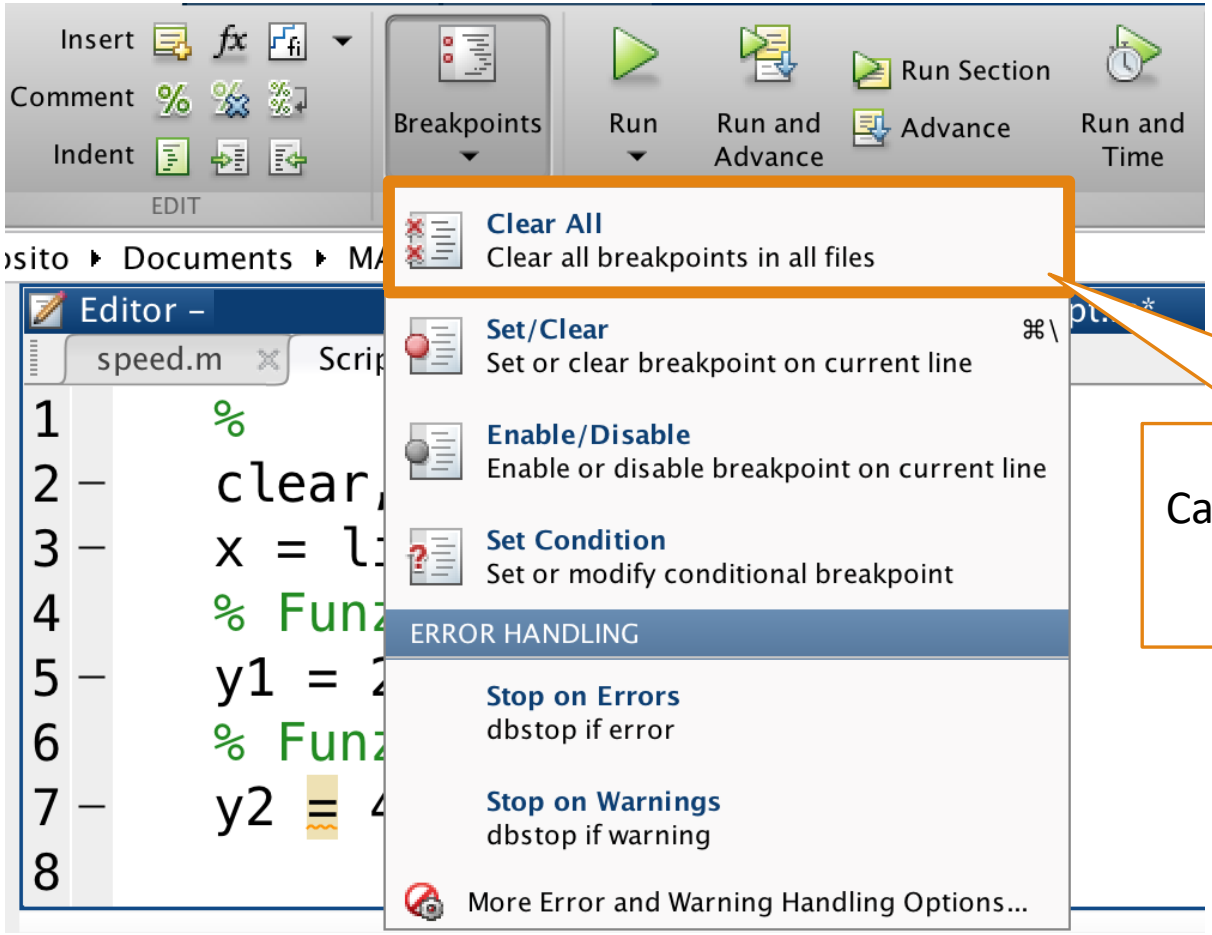
- Un *breakpoint* può essere considerato (valutato) solo al verificarsi di una determinata condizione (preimpostata), in base ai valori correnti delle variabili
  - La condizione è valutata prima dell'esecuzione dell'istruzione definita alla riga dove è stato inserito il *breakpoint*

Debugging – 6/7



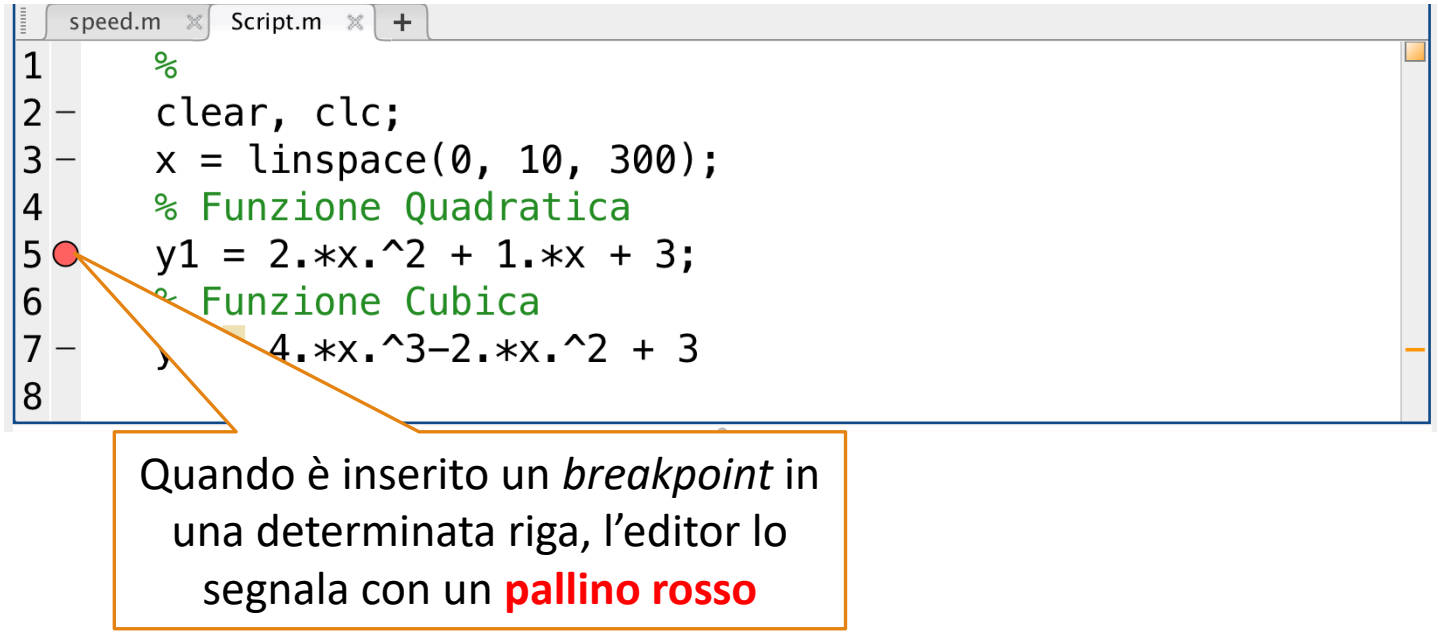
MATLAB fornisce diverse strategie per la gestione di eventuali errori di programmazione, che possono presentarsi durante l'esecuzione di un programma

Debugging – 6/7



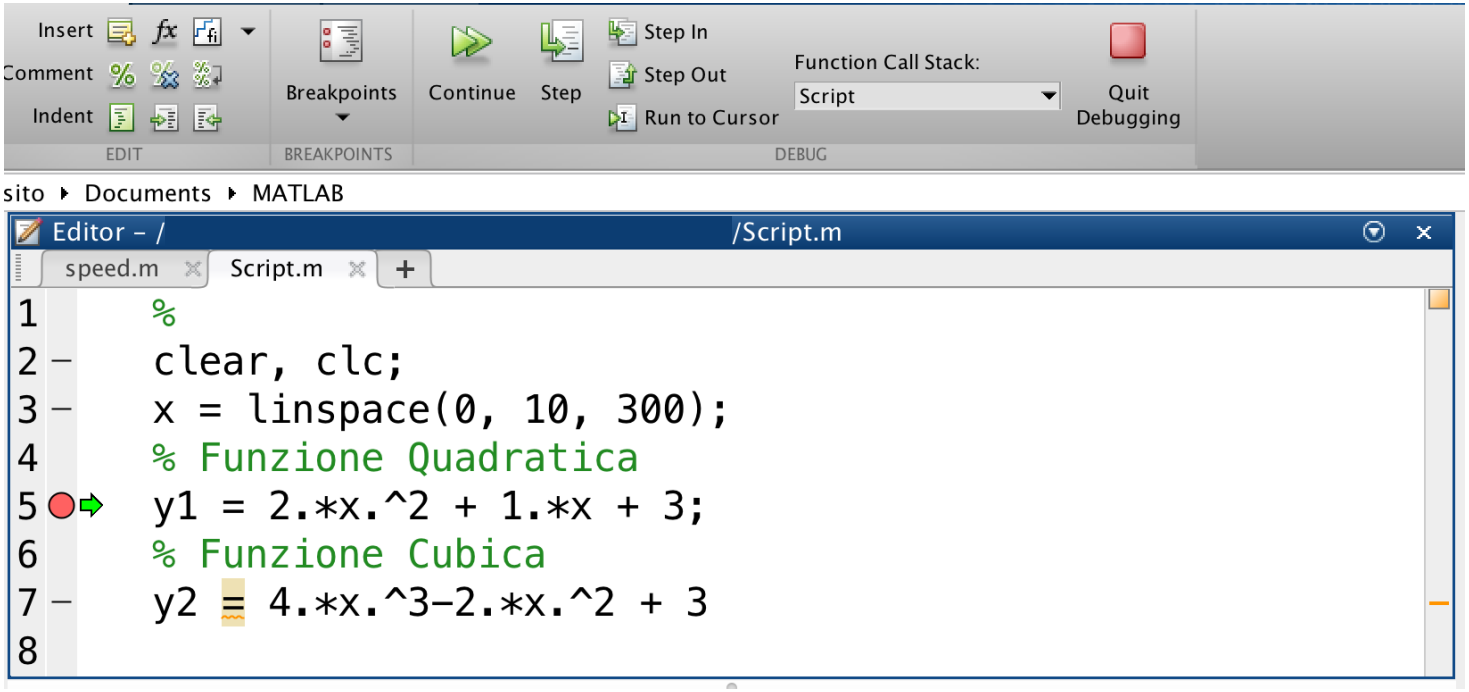
Cancella tutti i *breakpoints* nel programma

Debugging – 7/7



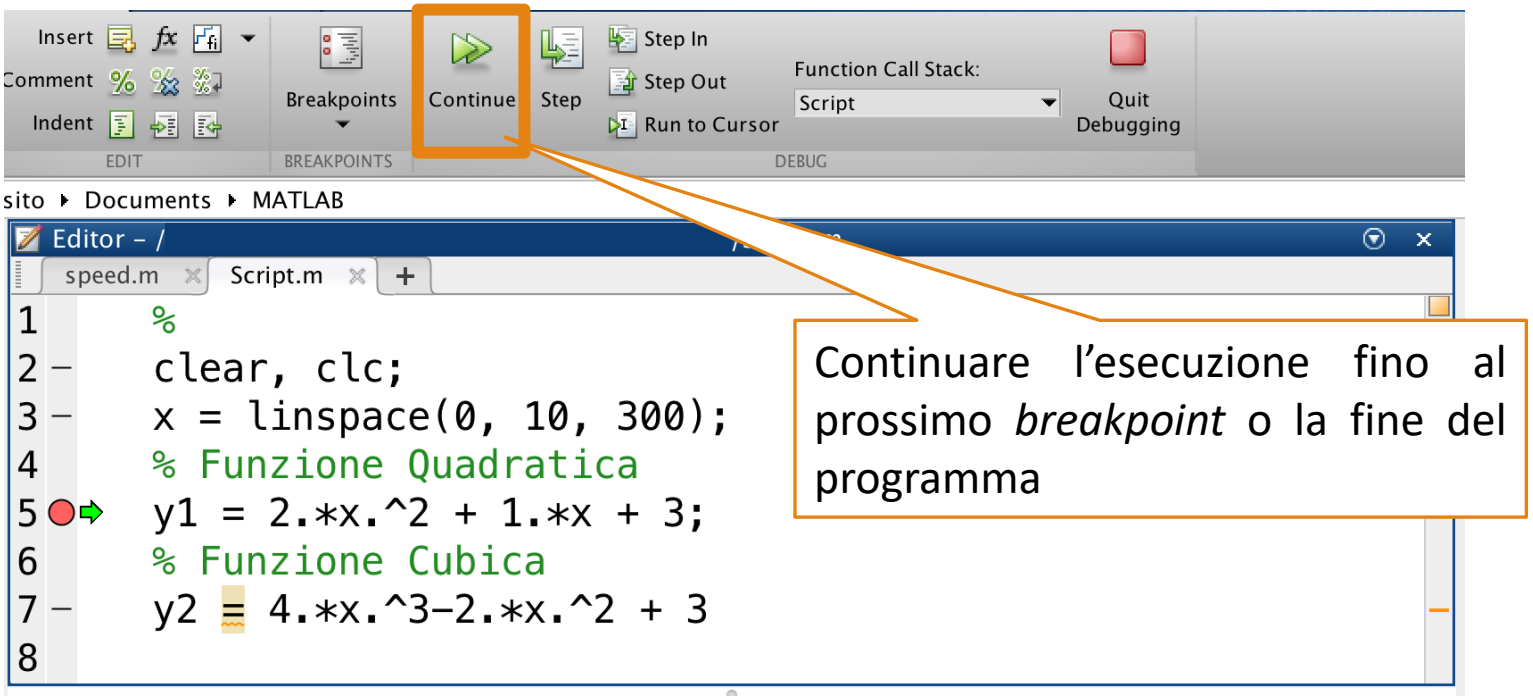
- Se scegliamo di eseguire la suddetta porzione di codice, noteremo che al punto del *breakpoint* si avrà un'interruzione temporanea del programma

Debugging – 7/7



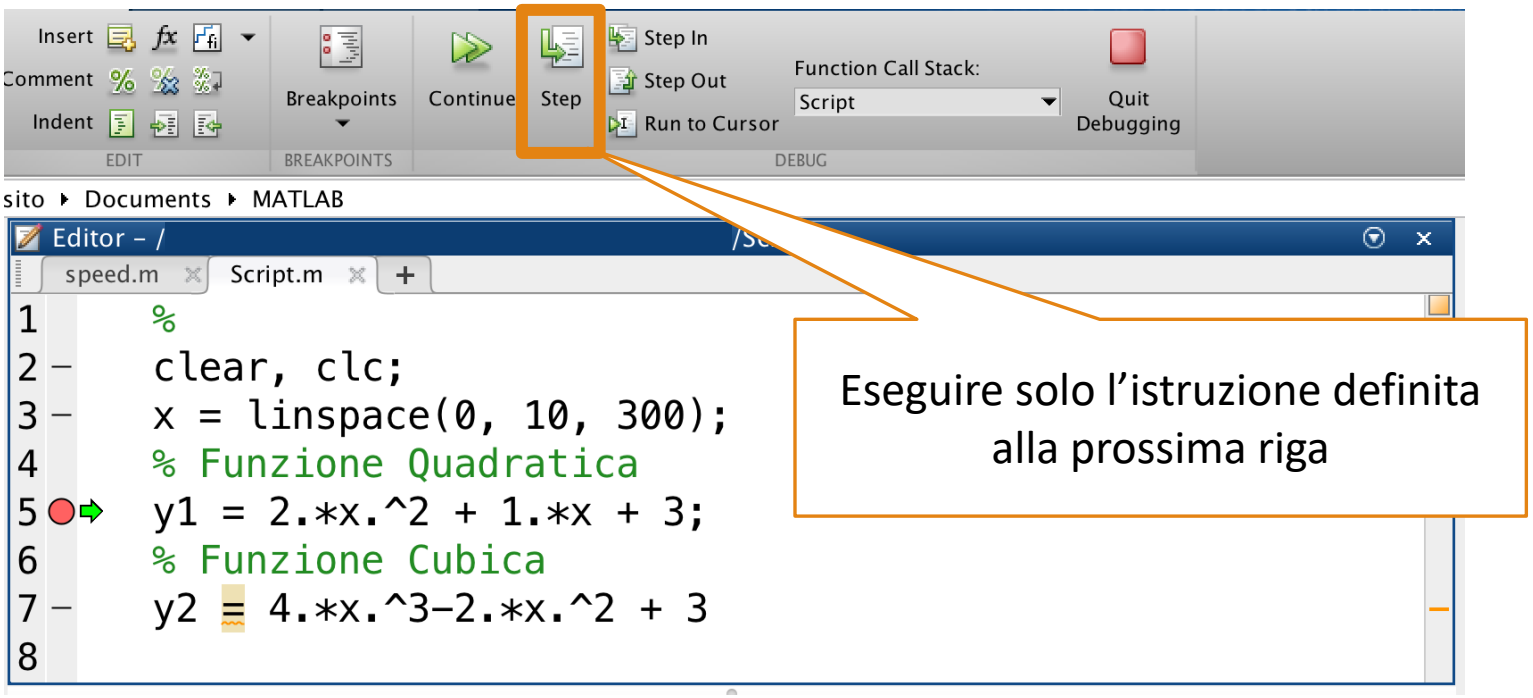
- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Debugging – 7/7



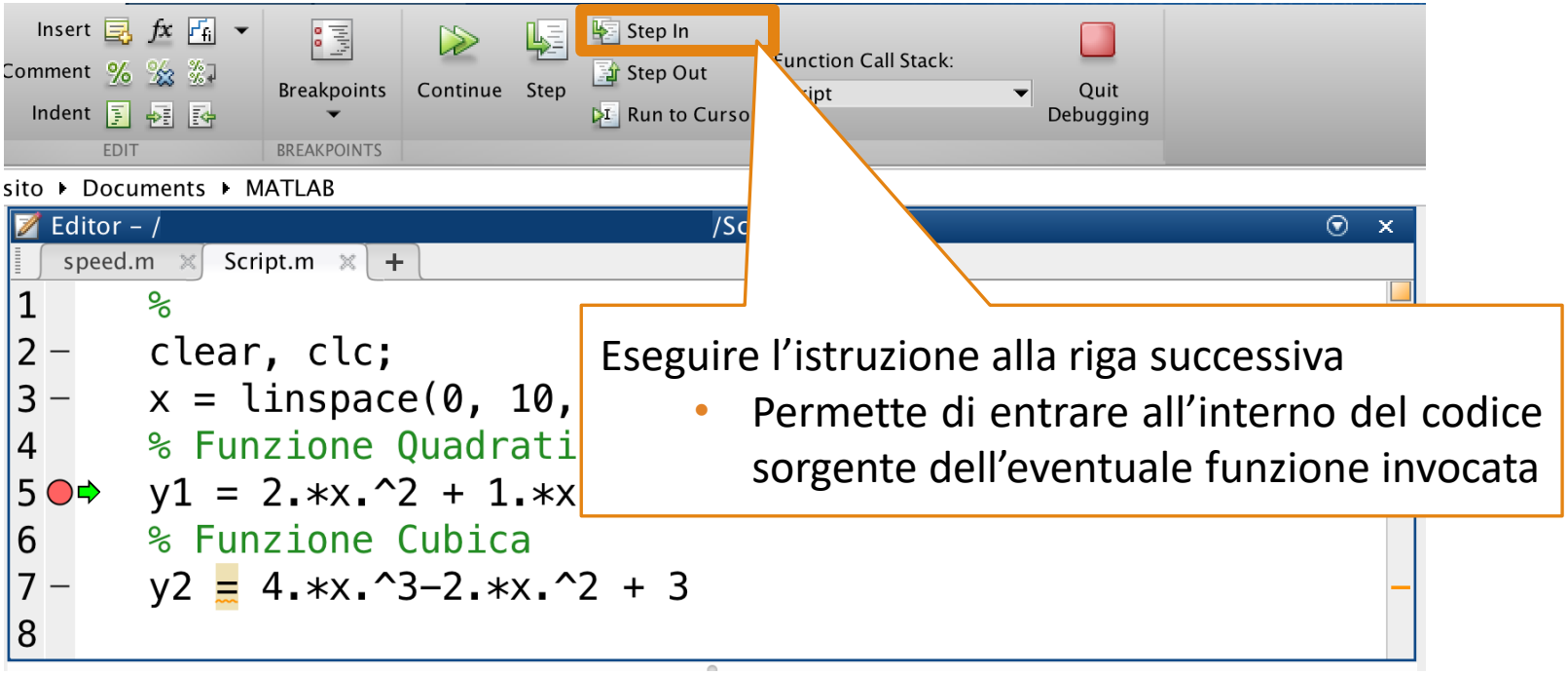
- Quando l'esecuzione del programma si interrompe su un breakpoint
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Debugging – 7/7



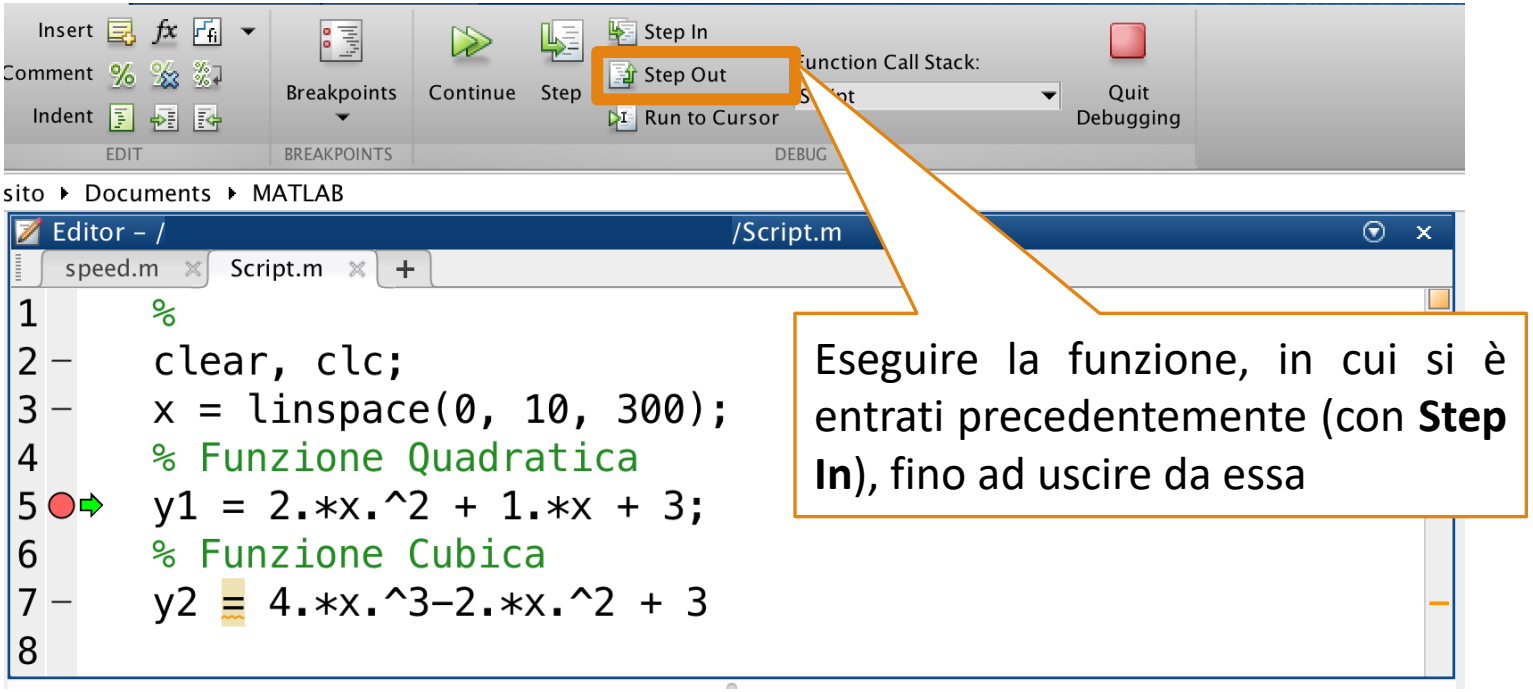
- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Debugging – 7/7



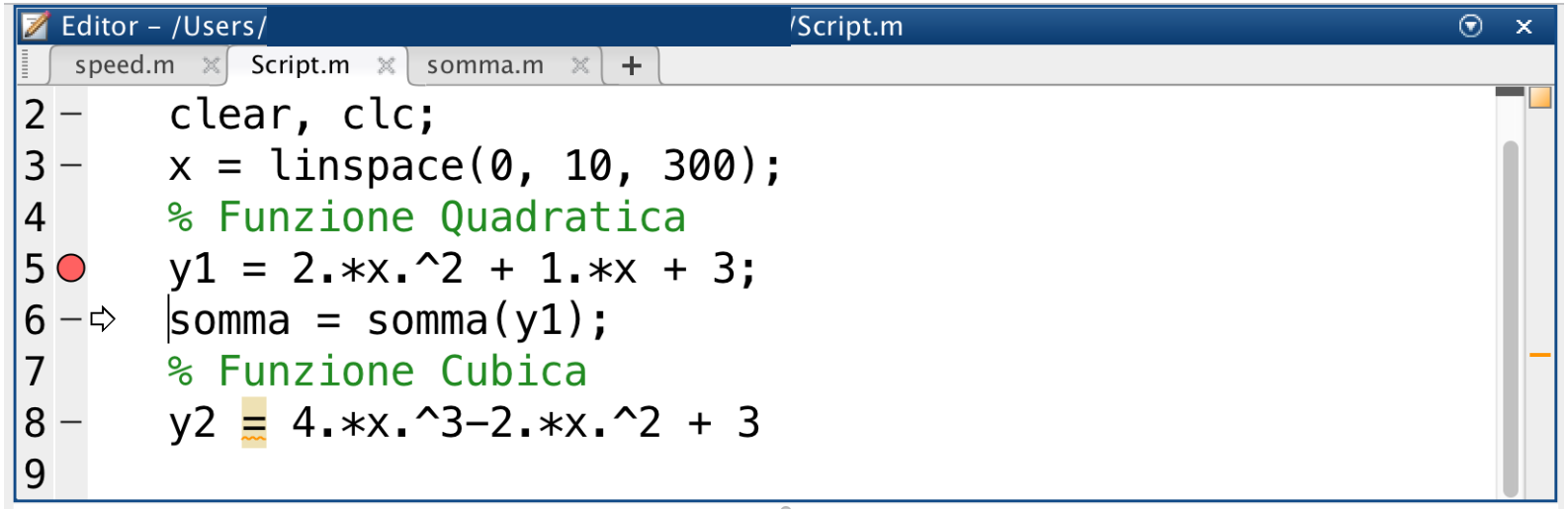
- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Debugging – 7/7

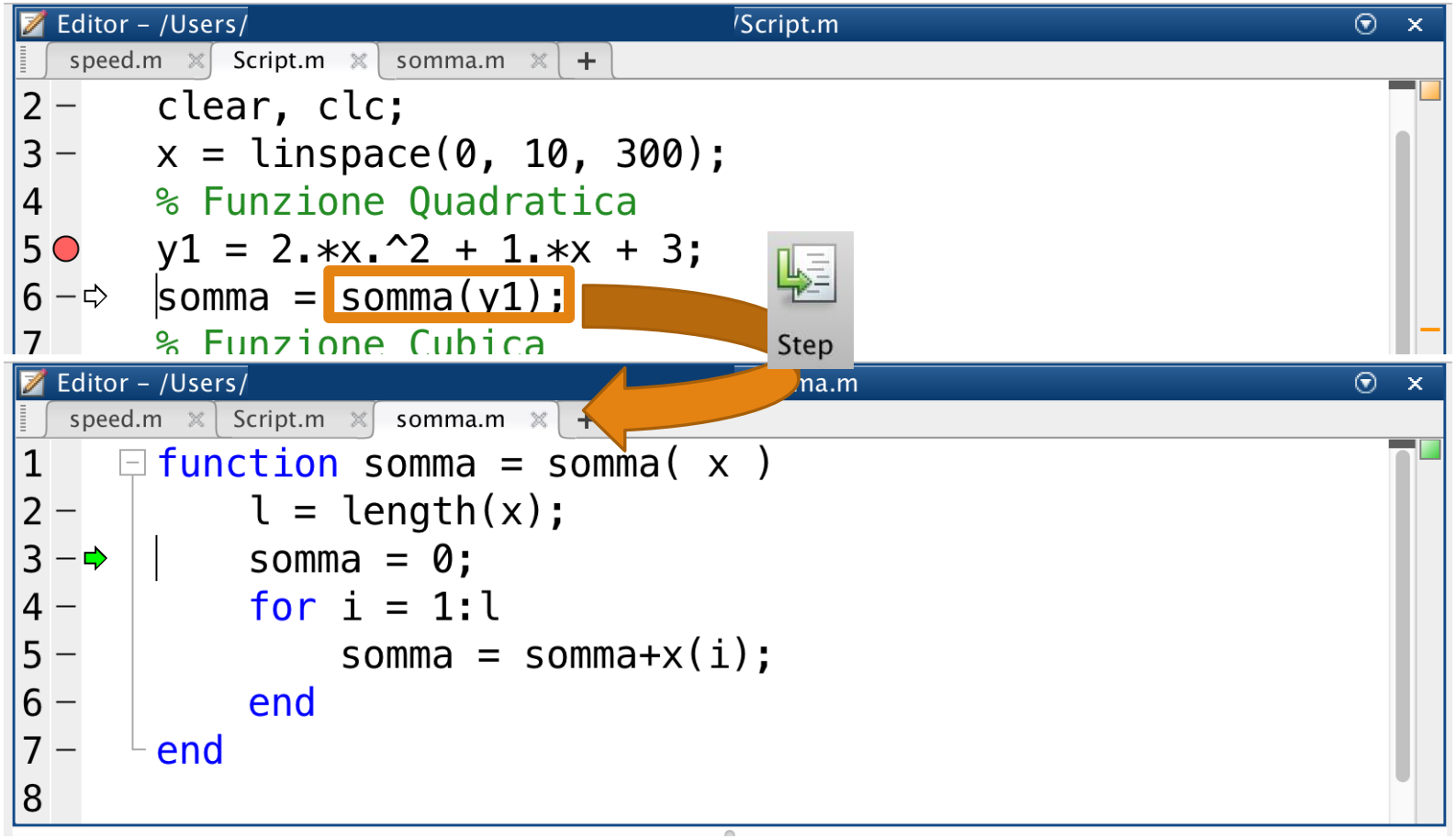


- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

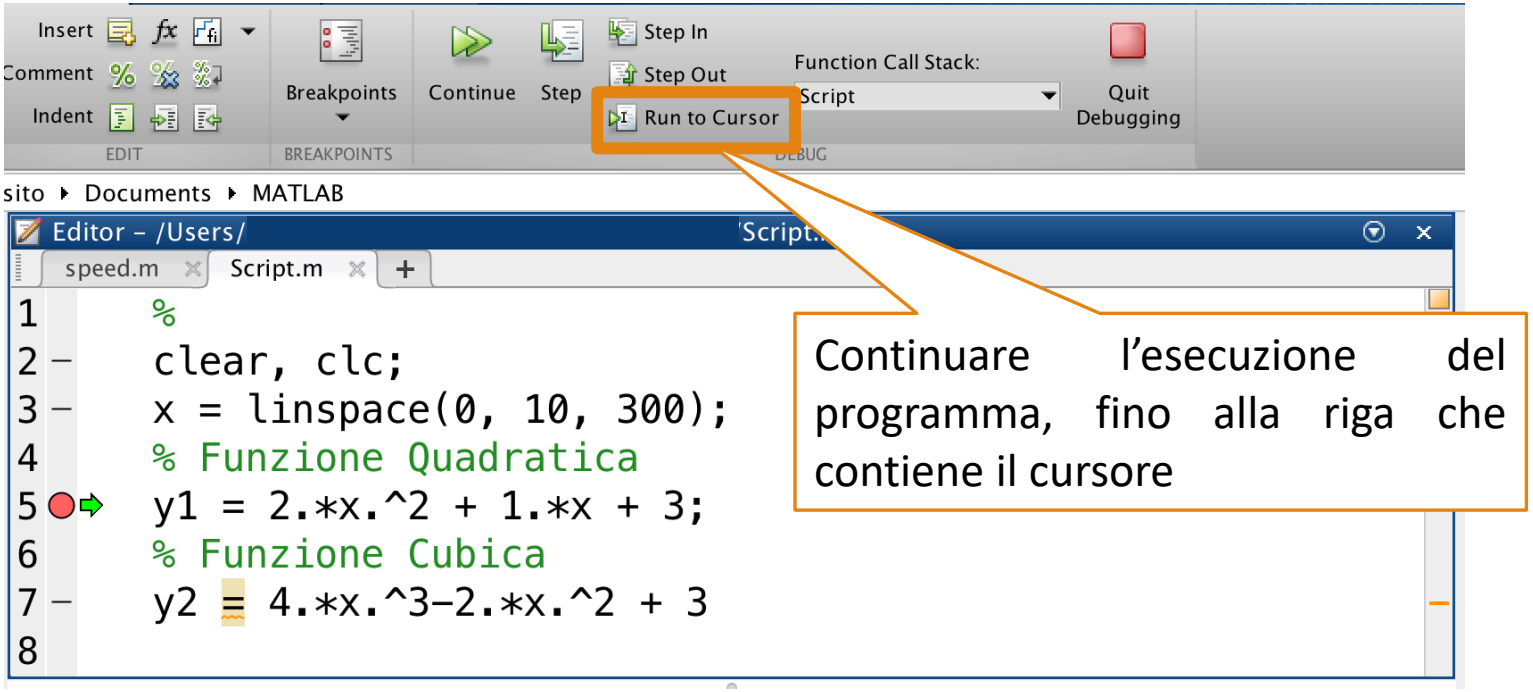
Debugging – 7/7



Debugging – 7/7

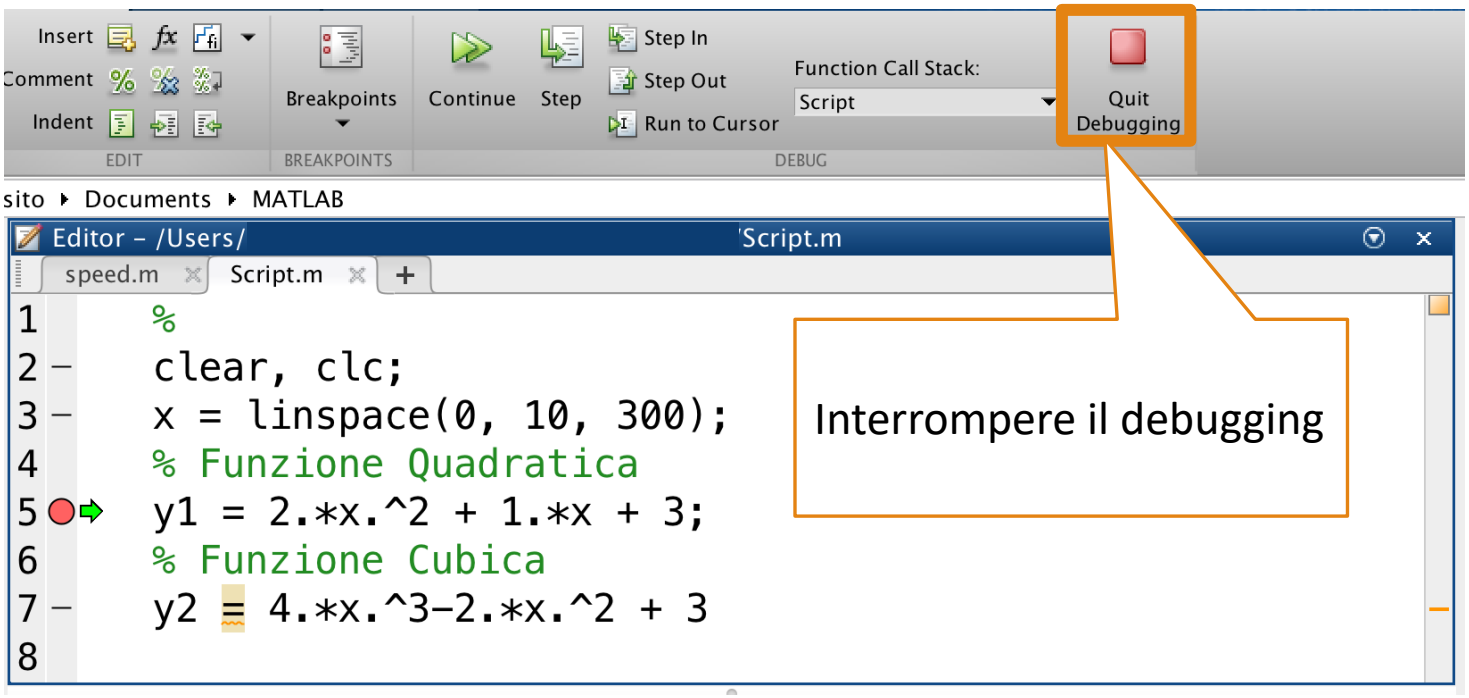


Debugging – 7/7



- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Debugging – 7/7



- Quando l'esecuzione del programma si interrompe su un *breakpoint*
  - MATLAB fornisce al programmatore diverse azioni da poter svolgere

Riferimenti

- Capitolo 4
  - Paragrafo 8 [**Debugging dei programmi di MATLAB**]