

Simulazione dei Sistemi dinamici con Matlab-Simulink

Funzioni

Ing. Alessandro Pisano
apisano@unica.it

Function files

Sono dei particolari files matlab che, a differenza dagli script, restituiscono una o più variabili in uscita che dipendono da una o più variabili passate alla funzione come parametri di ingresso.

Se non specificato diversamente, un Function File lavora esclusivamente su variabili locali create e distrutte alla attivazione ed alla chiusura della funzione.

All'interno di un Function File NON sono accessibili le variabili del workspace.

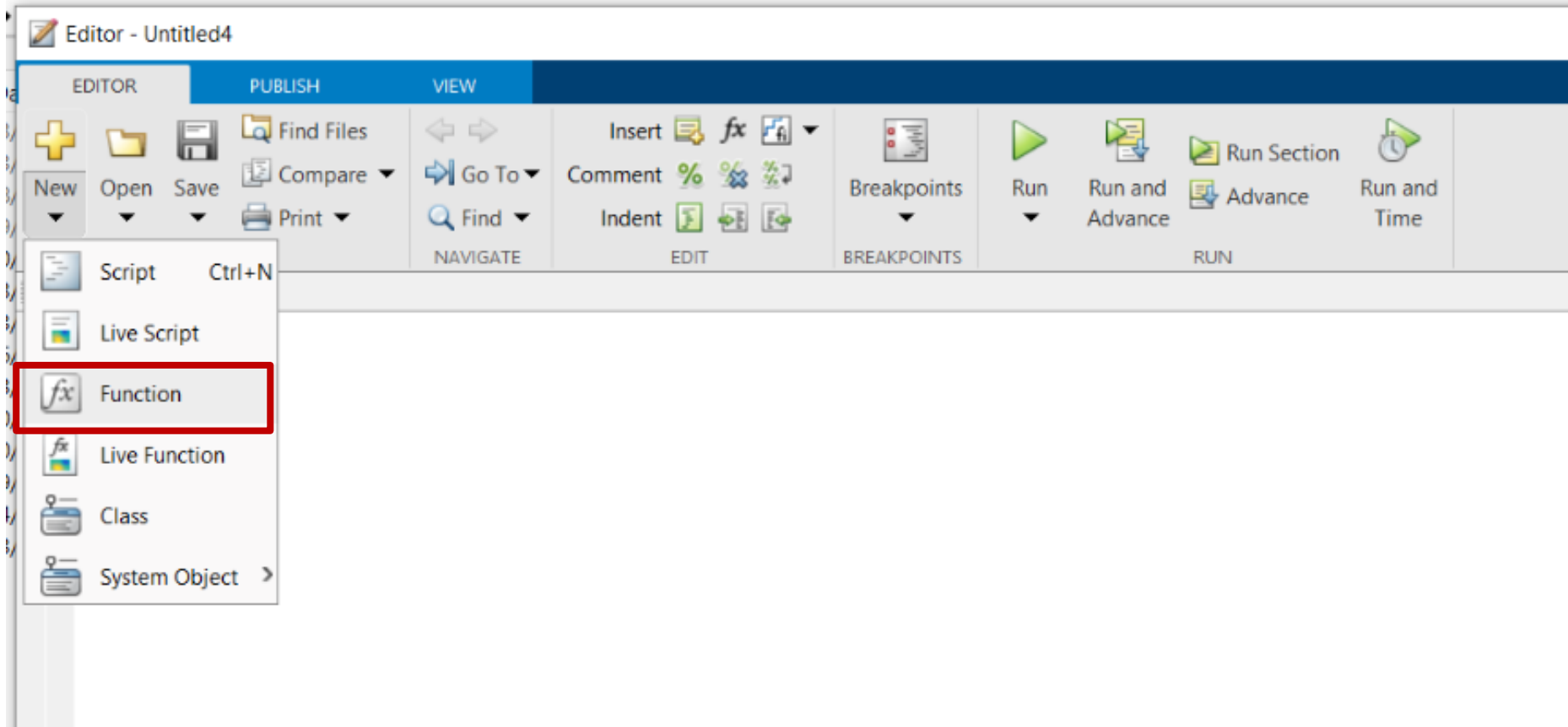
Si scrivono seguendo un formato standard, e **non** possono essere eseguite con il pulsante Save&Run dell'editor perche normalmente "attendono" argomenti in input.

Si avviano normalmente dal prompt di Matlab (o da uno script) con la sintassi:

```
[out1 out2]=nome_funzione(in1,in2,in3)
```

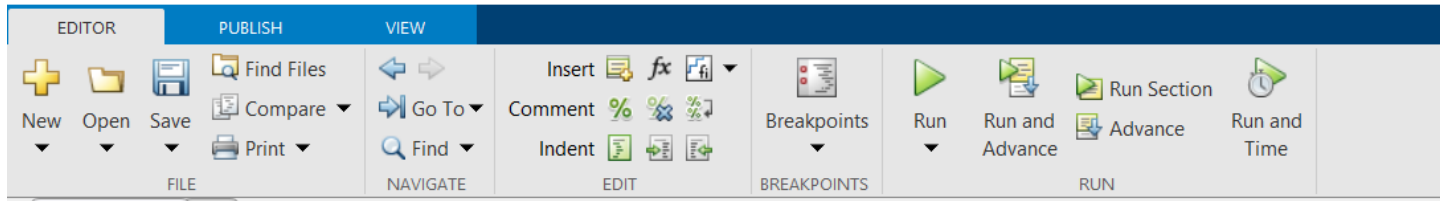
E' una funzione che accetta tre argomenti in ingresso e restituisce due argomenti in uscita. Sia gli ingressi che le uscite possono essere variabili di qualunque tipo (array, stringhe di testo,...).

Selezionare dall'editor: New->Function



Template di un function file

Editor - Untitled5*



```

1  function [outputArg1,outputArg2] = untitled5(inputArg1,inputArg2) → Riga di intestazione
2  %UNTITLED5 Summary of this function goes here
3  % Detailed explanation goes here → Help
4  outputArg1 = inputArg1;
5  outputArg2 = inputArg2; → Corpo della funzione
6  end
7
8

```

Il template si riferisce ad una funzione che accetta in ingresso due argomenti, e restituisce all'esterno altri due.

Esempio:

Scriviamo un function file che accetta in ingresso un vettore e ne restituisce il massimo, la media, e la varianza. Decidiamo che tale funzione si chiamerà «**maxmedvar**»

Come primo passo modifichiamo la prima riga della funzione, inserendo il numero corretto di ingressi e uscite, ed il nome della funzione

```
Untitled5* x +
1  function [ y1 , y2 , y3 ] = maxmedvar( v )
2
3  %UNTITLED5 Summary of this function goes here
4  %   Detailed explanation goes here
5
6  outputArg1 = inputArg1;
7  outputArg2 = inputArg2;
8  end
9
~
```

I nomi attribuiti a ingressi uscite sono completamente arbitrari.

Ora modifichiamo il corpo della funzione. Esso deve contenere le istruzioni che assegnano un valore a tutte le variabili di uscita. Tali istruzioni possono utilizzare esplicitamente i parametri passati in ingresso alla funzione. All'interno di un Function File, come già detto, **NON** sono accessibili le variabili del workspace.

```
Untitled5* x +
1 function [ y1 , y2 , y3] = maxmedvar( v )
2
3 %UNTITLED5 Summary of this function goes here
4 %   Detailed explanation goes here
5
6 y1=max(v) ;
7 y2=mean(v) ;
8 y3=var(v) ;
9 |
10 end
1
```

Per assegnare un valore alle tre uscite di questa funzione, utilizziamo le funzioni matlab `max`, `mean` e `var`

Per concludere, anche se non è indispensabile, si può inserire un help che spieghi come opera la funzione e cosa faccia. L'help può opzionalmente includere degli esempi di uso della funzione stessa.

```
function [ y1 , y2 , y3 ] = maxmedvar( v )  
  
% maxmedvar: questa funzione riceve un vettore in ingresso e ne  
% restituisce il massimo, la media e la varianza mediante tre variabili  
% distinte di uscita.  
% Esempio:      [mass med var]=maxmedvar(v)  
  
y1=max(v);  
y2=mean(v);  
y3=var(v);  
  
end
```

Ora salviamo il function file attribuendogli **lo stesso nome** della funzione che abbiamo inserito nella riga di intestazione. Questo è, tra l'altro, il nome che viene proposto da Matlab al momento del salvataggio.

Ora scriviamo uno script che impieghi la funzione che abbiamo appena salvato.

Lo script non farà altro che creare un vettore e passarlo in ingresso alla funzione.

```
Untitled6* x +
1 v=[-1 4 5 3 21 0];
2
3 [massimo,media,varianza]=maxmedvar(v)|
4
```

Output restituito dallo script nel prompt dei comandi

```
massimo =
    21
media =
    5.3333
varianza =
    64.2667
```

Digitando nel prompt dei comandi `help maxmedvar` compare:

```
>> help maxmedvar
```

```
maxmedvar: questa funzione riceve un vettore in ingresso e ne  
restituisce il massimo, la media e la varianza mediante tre variabili  
distinte di uscita.
```

```
Esempio: [mass med var]=maxmedvar(v)
```

Realizziamo la funzione in maniera diversa: ora facciamo in modo che restituisca all'esterno non più tre variabili distinte ma bensì un vettore di tre elementi le cui componenti siano, nell'ordine, il valore massimo, la media e la varianza del vettore in ingresso.

```
function [out] = maxmedvar2( v )

% maxmedvar: questa funzione riceve un vettore in ingresso e ne
% restituisce il massimo, la media e la varianza
% mediante un vettore di tre elementi.
% Esempio:
% out=maxmedvar(v)
% massimo=out(1);
% media=out(2);
% varianza=out(3);

out(1)=max(v);
out(2)=mean(v);
out(3)=var(v);

end
```

Esempio

Function file che riceve in ingresso due matrici quadrate di pari dimensione e restituisce il determinante e gli autovalori della matrice prodotto

```
function [determ autov] = mat1(A1,A2)
%mat1 funzione che riceve in input due matrici quadrate
% di pari dimensione e restituisce il determinante e gli
% autovalori della matrice prodotto
% Esempio di utilizzo:
%     [D A]=mat1(M1,M2)

determ=det(A1*A2);
autov=eig(A1*A2);
end
```

Esempio

Function file che riceve in ingresso una matrice quadrata A e restituisce il più grande ed il più piccolo autovalore della sua parte simmetrica $(A + A^T)/2$ (N.B. gli autovalori di una matrice simmetrica sono reali)

Esempio

Function file che riceve in ingresso una matrice quadrata A e restituisce il più grande ed il più piccolo autovalore della sua parte simmetrica $(A + A^T)/2$ (N.B. gli autovalori di una matrice simmetrica sono reali)

```
function [eigmin eigmax] = myfun2 (M)
%Questo file

Msimm=(M+M')/2;
eigmin =min(eig(Msimm));
eigmax = max(eig(Msimm));
end
```

La funzione può essere chiamata così:

```
[autovmin autovmax]=myfun2 (M)
```

Esempio

Si dispone di un file excel (DemograficaSardegna2019.xls) che contiene dei dati demografici. In particolare, tale file contiene il numero di abitanti, suddivisi per età e sesso, residenti in Sardegna al 1 gennaio 2019.

	A	B	C	D
1	Eta	Maschi	Femmine	
2	0	4788	4503	
3	1	5238	4878	
4	2	5445	5047	
5	3	5767	5309	
6	4	5876	5470	
7	5	6052	5705	
8	6	6406	5967	
9	7	6694	6333	
10	8	6950	6451	
11	9	7017	6412	
12	10	6943	6549	
13	11	6875	6559	
14	12	7036	6501	
15	13	7004	6545	

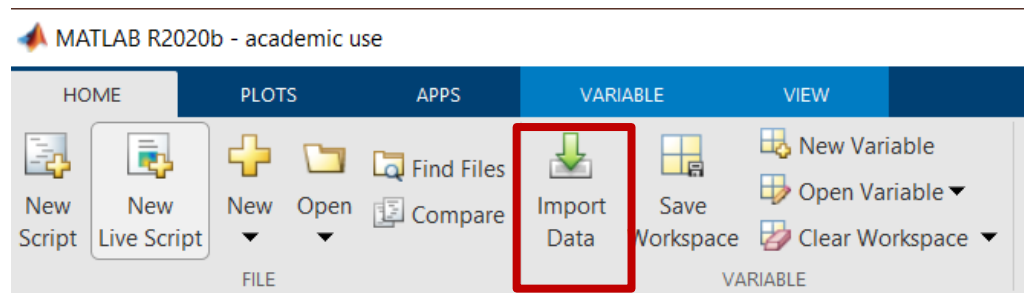
⋮

	A	B	C	D
97	95	406	983	
98	96	295	772	
99	97	178	531	
100	98	120	419	
101	99	69	228	
102	> 100	90	322	
103				
104				

Si desidera realizzare un function file che elabori tali dati e **restituisca il numero di residenti maschi e femmine appartenenti ad una certa fascia di età**.

Il function file riceve in ingresso gli estremi inferiore e superiore dell'intervallo di età desiderato.

Il **primo passo** è acquisire i dati in Matlab. Utilizziamo a tal fine la funzionalità «Import Data», accessibile tramite opportuno pulsante nella finestra principale di Matlab



Nel momento in cui si preme il pulsante Import Data si apre una finestra che consente di selezionare il file dal quale si desidera importare i dati. Selezionato il file excel di nostro interesse si visualizza quanto segue:

DemograficaSardegna2019			
Eta	Maschi	Femmine	
Number	Number	Number	
1	Eta	Maschi	Femmine
2	0	4788	4503
3	1	5238	4878
4	2	5445	5047
5	3	5767	5309
6	4	5876	5470
7	5	6052	5705
8	6	6406	5967
9	7	6694	6333
10	8	6950	6451
11	9	7017	6412
12	10	6943	6549
13	11	6875	6559
14	12	7036	6501
15	13	7081	6545
16	14	6955	6449
17	15	7207	6709
18	16	7017	6671
19	17	7196	6822
20	18	7651	6925
21	19	7777	6771
22	20	7709	6632
23	21	7725	6745

Desideriamo importare all'interno di una matrice i dati contenuti nelle colonne «Maschi» e «Femmine».

In aggiunta al fatto che i dati presenti nella prima colonna sono ridondanti, l'ultimo elemento della colonna «Età» non è un numero («>100») e come tale non può essere importato in un array numerico.

Import - C:\FilesAlessandro\Didattica\Simulazione dei sistemi dinamici con Matlab Sim

IMPORT VIEW

Range: B2:C102 Output Type: Numeric Matrix

Variable Names Row: 1

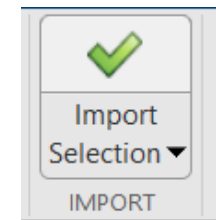
SELECTION IMPORTED DATA

DemograficaSardegna2019.xlsx

	A	B	C
	DemograficaSardegna2019		
1	Eta	Maschi	Femmine
2	0	4788	4503
3	1	5238	4878
4	2	5445	5047
5	3	5767	5309
6	4	5876	5470
7	5	6052	5705
8	6	6406	5967
9	7	6694	6333
10	8	6950	6451
11	9	7017	6412
12	10	6943	6549
13	11	6875	6550

Selezioniamo quindi, nella casella «Range», l'intervallo di celle che desideriamo acquisire, e nella casella «Output Type» selezioniamo «Numeric Matrix»

Fatto ciò, premendo il pulsante «Import Selection» collocato sulla destra:



viene creata nel workspace una variabile matriciale che contiene i dati selezionati e che ha lo stesso nome del file excel acquisito.

Abbiamo creato una matrice con due colonne e 101 righe.

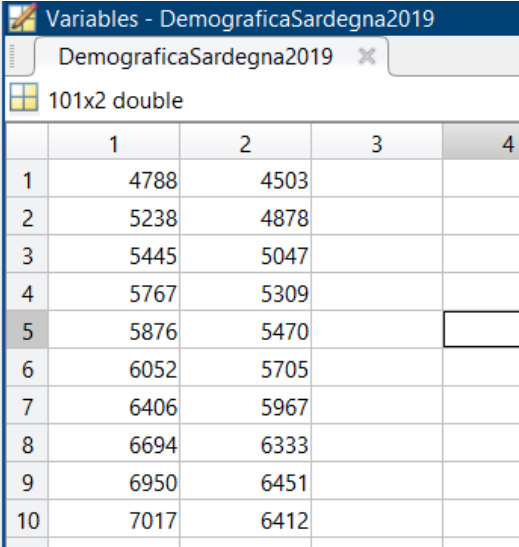
Facciamo doppio click nella sottofinestra «workspace» sulla variabile appena creata.

La prima riga contiene il numero di residenti aventi meno di 1 anno

La seconda riga contiene il numero di residenti aventi 1 anno

La terza riga contiene il numero di residenti aventi 2 anni.

I dati corrispondenti ad un valore generico di età pari ad n ($n=0,1,2,3,\dots$) si troveranno quindi nella riga $n+1$ della matrice



	1	2	3	4
1	4788	4503		
2	5238	4878		
3	5445	5047		
4	5767	5309		
5	5876	5470		
6	6052	5705		
7	6406	5967		
8	6694	6333		
9	6950	6451		
10	7017	6412		

Ora realizziamo la funzione.

```
function [M,F] = demografic(A,etamin,etamax)

maschi=A(etamin+1:etamax+1,1);
femmine=A(etamin+1:etamax+1,2);

M=sum(maschi);
F=sum(femmine);

end
```

La variabile `maschi` generata nella prima riga del corpo della funzione sarà un vettore che contiene gli elementi della prima colonna della matrice aventi indice compreso fra `etamin+1` ed `etamax+1`. Analogamente per quanto concerne la variabile `femmine` che estrare un sottoinsieme degli elementi della seconda colonna della matrice.

Per generare i dati da produrre in uscita devono successivamente essere sommati fra loro (funzione `sum`) tutti gli elementi dei vettori `maschi` e `femmine`

Ora realizziamo un semplice script che richiama la funzione creata, e la utilizza per determinare il numero di residenti di età compresa fra 0 e 18 anni.

```
A=DemograficaSardegna2019;  
[Mas,Fem]=demografic(A,0,18)
```

Output:

```
Mas =  
    124194  
  
Fem =  
    115805
```

Accesso di un function file alle variabili del workspace

L'unico modo attraverso il quale una funzione può accedere a variabili del workspace è che tali variabili siano definite come **globali** (`global`)

Le variabili devono essere definite come globali esternamente a tutte le funzioni.

```
global var1 var2 var3
```

Se una funzione desidera accedere ad alcune delle variabili, ad es. `var1` e `var3`, allora tali variabili devono essere ridefinite come globali **anche all'interno del corpo della funzione**

```
global var1 var3
```

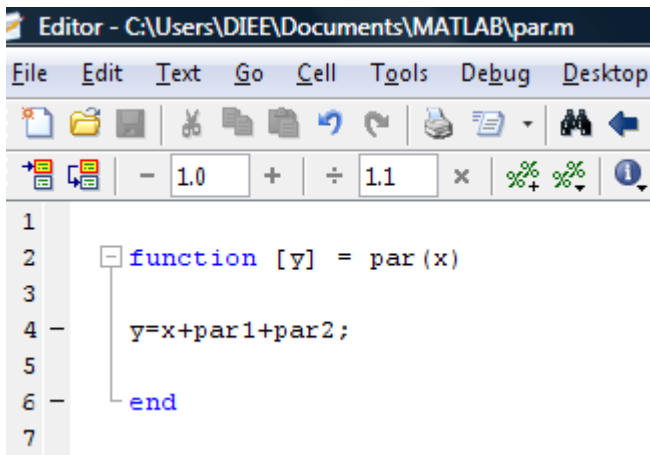
Esempio

Mediante uno script si caricano nel workspace i valori di determinati parametri

```
clear all  
clc
```

```
par1=1;  
par2=2;
```

Il seguente function file prova ad accedere ai parametri par1 e par2

The image shows a screenshot of the MATLAB Editor window. The title bar reads "Editor - C:\Users\DIEE\Documents\MATLAB\par.m". The menu bar includes "File", "Edit", "Text", "Go", "Cell", "Tools", "Debug", and "Desktop". Below the menu bar is a toolbar with various icons for file operations and editing. A numeric keypad is visible with values "1.0" and "1.1". The main editing area shows a function definition starting at line 2: "function [y] = par(x)", followed by "y=x+par1+par2;" on line 4, and "end" on line 6. Line numbers 1, 3, 5, and 7 are visible on the left margin.

```
1  
2 function [y] = par(x)  
3  
4 - y=x+par1+par2;  
5  
6 - end  
7
```

```
function [y] = par(x)
```

```
y=x+par1+par2;
```

```
end
```

Eseguendolo nel prompt di Matlab si ottiene un messaggio di errore

```
>> a=par(3)
??? Undefined function or variable 'par1'.

Error in ==> par at 4
y=x+par1+par2;
```

Le variabili par1 e par2 devono essere dichiarate globali, sia nello script di definizione dei parametri ...

```
clear all
clc
global par1 par2
par1=1;
par2=2;
```

... che nel corpo del function file.

```
function [y] = par(x)
global par1 par2
y=x+par1+par2;
end
```

Ora il function file viene eseguito correttamente

```
>> a=par(3)

a =

     6
```

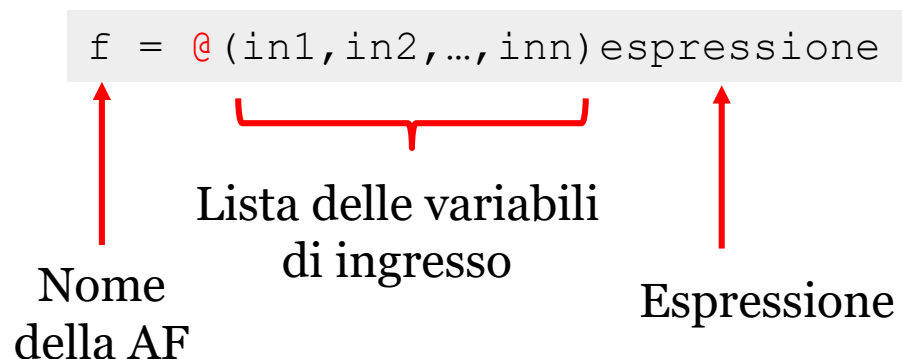
Anonymous functions (AF)

E' a tutti gli effetti un function file che non viene salvato in un m-file ma risiede in una variabile. E' quindi locale alla sessione di lavoro corrente e semplifica progetti complessi riducendo il numero complessivo di files del progetto.

Risulta essere una espressione che può avere un numero arbitrario di variabili di ingresso ma un'unica uscita. Tale uscita può però essere costituita da un vettore, o da una matrice, quindi è possibile fare in modo che la AF restituisca un numero arbitrario di quantità di interesse.

Una AF sarà definita normalmente all'interno di uno script o di un function files.

La **sintassi** per creare una AF è la seguente



Esempio Scrivere una AF che implementa la seguente funzione:

$$g(x) = x^2 + 3x + 4$$

La AF sarà la seguente

```
g = @(x) x^2+3*x+4;
```

La utilizzeremo, ad esempio in uno script, come segue:

```
a1=g(4)  
a2=g(10)
```

Ottenendo il seguente output

```
a1 =  
    32  
  
a2 =  
   134
```

Una AF può avere più di un parametro in ingresso

Esempio Scrivere una AF che implementa la funzione:

$$g(x, n) = x^n$$



```
g = @(x, n) x^n;
```

Esempio Scrivere una AF che implementa la funzione:

$$f(x, y) = x^2 + y^2 + 2xy + y + 5$$



```
f = @(x, y) x^2+y^2+2*x*y+y+5;
```

Una AF può ricevere in ingresso vettori o matrici

Esempio Scrivere una AF che riceve in input due vettori e restituisce uno scalare che contiene la somma di tutti gli elementi del primo vettore più il primo elemento del secondo vettore:



$$g = @(v, w) (\text{sum}(v) + w(1));$$

Una AF può restituire in uscita vettori o matrici

Esempio Scrivere una AF che riceve in input uno scalare x e restituisce in uscita

la matrice $\begin{bmatrix} 1 & x & x^2 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix}$

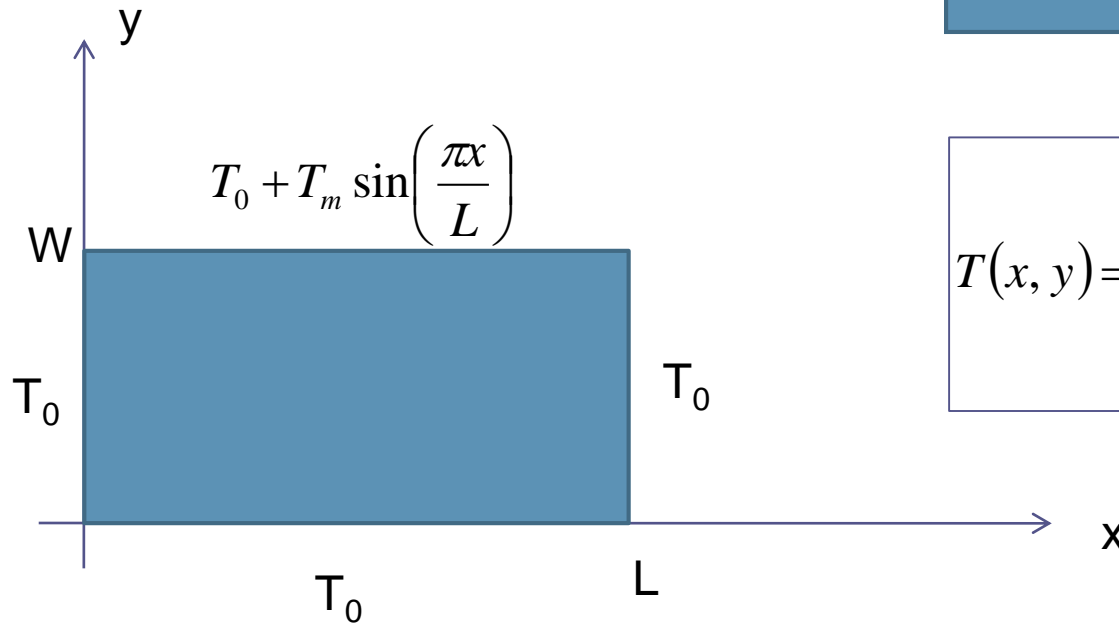


$$g = @(x) [1 \ x \ x^2; 0 \ 1 \ x; 0 \ 0 \ 1];$$

Una AF può impiegare variabili presenti nel workspace

Esempio

Scrivere una AF che implementi la distribuzione di temperatura a regime in una sezione di una sbarra metallica a sezione rettangolare di lunghezza infinita con temperatura nel bordo imposta dall'esterno.



$$T(x, y) = T_0 + T_m \sin\left(\frac{\pi x}{L}\right) \frac{\sinh\left(\frac{\pi y}{L}\right)}{\sinh\left(\frac{\pi W}{L}\right)}$$

$$T(x, y) = T_0 + T_m \sin\left(\frac{\pi x}{L}\right) \frac{\sinh\left(\frac{\pi y}{L}\right)}{\sinh\left(\frac{\pi W}{L}\right)}$$

W=0.6;

L=0.6;

T0=25;

Tm=10;

`f=@(x,y) (T0+Tm*sin(pi*x/L)*sinh(pi*y/L)/sinh(pi*W/L));`

P1=f(0.3,0.01)

P2=f(0.3,0.3)

P3=f(0.3,0.59)

