

## Introduzione 2

### Comandi di grafica:

Andiamo a vedere alcuni comandi di grafica

```
>> xx = linspace(0,1,0.11) (11 nodi equidistanti tra 0 ed 1)
ans =
    0
   0.1
   ...
   ...
   ...
   0.9
    1
```

```
>> yy=sin(xx);
>> plot(xx,yy)
```

Se voglio disegnare piú grafici contemporaneamente di due colori diversi

```
>> plot(x,y,'r',xx,yy,'k') (r = rosso, k = nero)
```

**Osservazione:** Usare i tasti direzionali della tastiera per tornare alle vecchie istruzioni.

```
>> plot(x,y,'or',....) (disegna dei cerchietti)
>> plot(x,y,'o-r',....) (cerchietti uniti da linee)
>> plot(x,y,'+k',....) (crocette)
```

Vediamo adesso come si scrive sulle interfacce grafiche

```
>> plot(x,y,xx,yy)
>> title('grafici di seno e coseno') (titolo del grafico)
>> xlabel('asse x') (scrive sull'asse delle x)
```

Ancora un esempio da linea di comando

```
>> clear
>> x=linspace(0,2,201);
>> y4=exp(-x).*sin(4*x);
>> plot(x,y4);
```

Vediamo adesso come si uniscono piú finestre grafiche contemporaneamente, utilizzeremo il comando "subplot":

```
>> subplot(2,2,1) %produce una finestra 2x2 in cui al primo
                  %posto posizionera' il seguente grafico
>> plot(x,y4); title('caso n = 4');
>> subplot(2,2,2); % questo grafico nel posto 1,2
>> plot(x,y5); title('caso n = 5');
```

### Programmazione in Matlab: Il comando FOR.

Voglio calcolare la somma degli elementi di un vettore

```
>> x = [ ..... ]; %vettore x per elencazione
>> sum = 0;
>> for i=1:9
        sum=sum+x(i); %attenzione: usare spesso il ";"
    end
```

**Esempio:** Procedura che calcola la media delle componenti di un vettore.

Andiamo a creare un nuovo m-file

```
n=length(x);      %dimensione del vettore x
sum = 0;
for i=1:n
    sum=sum+x(i);
end
m=sum/n;
```

Salviamo il file come "media.m" ed eseguiamolo:

```
>> clear
>> x = [ ..... ]; %vettore x per elencazione
>> media
>> m
    m =
        .....
```

**Esempio:** Costruzione della matrice di Hilbert.

La matrice di Hilbert é una matrice i cui elementi verificano la proprietá

$$a_{i,j} = \frac{1}{i+j-1}.$$

Costruiamo il file "hilbert.m"

```
for i=1:N
    for j=1:n
        A(i,j)=1/(i+j-1);
    end
end
```

Facciamo girare il file

```
>> clear
>> n=6;
>> N=6;
>> hilbert
>> A
    A =
        .....
```

é anche possibile dare la dimensione della matrice con il comando "input"

```
n=input('dammi la dimensione n = '); % scrive a schermo la stringa e
                                       % una volta digitato il valore lo salva
                                       % nella variabile n
N=n;
.....
```

Passando nella Command Window

```
>> clear
>> hilb
>> dammi la dimensione n = 5      % il 5 e' stato digitato
```

## Il comando IF:

Voglio rendere positive tutte le componenti negative di un vettore e lasciare inalterate le altre

```
>> x = [.....]      %vettore ad 8 componenti
>> for i=1:8
        if x(i)<0
            x(i)=-x(i);
        end
    end
>> x
x =
    .....
```

**Esempio:** Ricerca della massima componente di un vettore a componenti positive

Costruiamo un nuovo m-file e salviamolo come "norma.m"

```
n=length(x);
for i=1:n
    if x(i)<0
        x(i)=-x(i);
    end
end
max=x(1);
for i=2:n
    if x(i)>max
        max=x(i);
    end
end
max
```

Andiamo nella Command Window

```
>> clear
>> x = [.....]
>> norma
max =
    .....
```

**Esempio:** Costruzione di una matrice quadrata con tutti "4" sulla diagonale principale, "-1" sulle due diagonali secondarie, "0" altrove:

```
clear
n=input('dammi la dim n = ')
for i=1:n
    for j=1:n
        if i==j
            A(i,j)=4;
        else if (j==i-1)|(j==i+1)    % il simbolo "|" rappresenta l' "OR" logico
            A(i,j)=-1;
        else
            A(i,j)=0;
        end
    end
end
end
```

Che poi salveremo come "matrice.m" eseguendola quindi dalla Command Window.